MASTER EN SCIENCES INFORMATIQUES

Conception d'un logiciel calculant le risque de contracter un infarctus du myocarde

Benhayoun, Jawad

Award date:
1990

Awarding institution:
Universite de Namur

FACULTES UNIVERSITAIRES
N.D. DE LA PAIX
NAMUR

INSTITUT D'INFORMATIQUE

CONCEPTION D'UN LOGICIEL CALCULANT LE RISQUE DE CONTRACTER UN INFARCTUS DU MYOCARDE

Institut d'Informatique
ANNEE 1989-1990
Promoteur: J. FICHEFET

Mémoire présenté pour
l'obtention du titre
de Licencié et Maître
en Informatique par

BENHAYOUN Jawad

RUE GRANDGAGNAGE, 21, B - 5000 NAMUR (BELGIUM)

A

La Mémoire de mon père,

à

ma mère,
mes beaux-parents,
mon épouse Najia,
mes enfants Othmane et Amine,
et toute notre famille.

REMERCIEMENTS

Je voudrais remercier ici les personnes qui m'ont apporté leur aide pour la bonne réalisation de mon travail de fin d'étude.

Tout spécialement je remercie la clinique Saint Camille de Namur qui m'a acceuilli et plus particulièrement le Docteur Ph Beyne pour ses conseils et explications médicales.

Je remercie aussi Madame M. NOIRHOMME et Monsieur J.FICHFET pour leurs explications scientifiques plus que nécessaire pour la réalisation de ce mémoire.

Je remercie aussi tous les membres de l'Institut d'informatique qui m'ont permis de m'instruire dans de bonnes conditions.

TABLE DES MATIERES

CHAPITRE I DESCRIPTION DU PROBLEME

I.1INTRODUCTION

L'infarctus du myocarde est une lésion définitive, ou nécrose, d'un tissu ou d'un organe provoquée par l'obstruction de l'artère qui assure son irrigation. L'abolition brutale de l'apport de sang entraîne la mort à brève échéance des cellules. Dans le cas de l'infarctus du myocarde, qui atteind donc le muscle cardiaque, la diminution de l'apport sanguin est due à un rétrécissement ou à une obstruction d'une des artères qui irriguent le coeur, les artères coronaires. Dans quarante pour cent des cas, l'infarctus du myocarde s'accompagne de complications mortelles, dont la moitié survient dans la première heure de l'infarctus. Parmi les patients qui survivent au trentième jour d'un infarctus 80 °/° d'entre eux peuvent reprendre leur activité dans les mois qui suivent. Sur ces 80°/°, le quart meurt cinq ans plus tard et la moitié dix ans plus tard. Entre temps, nombreux sont ceux qui souffrent d'angine de poitrine ou d'insuffisance cardiaque justifiant des hospitalisations itératives. Ce sont surtout les hommes à l'age adulte et les vieillards qui sont attaqués (cinq fois plus que les femmes).

La lésion de base responsable de l'infarctus du myocarde(IM) consiste dans le

développement anormal, dans la paroi des artères coronaires, de plaques formées de graisses, l'athérome.

A l'état normal, la paroi d'une artère est composée de trois couches distinctes, dont la plus interne, est, l'intima directement en contact avec le sang. L'athérome est une lésion de l'intima. En réponse à une agression de la paroi artérielle, les cellules endothéliales, ainsi que les cellules musculaires, les plus internes, se mettent à proliférer pour recouvrir l'intima altérée. Cette prolifération s'accompagne d'un épaississement de l'intima et très vite les cellules se chargent de lipides (graisses), de cristaux de cholestérol, et se calcifient. Il se constitue ainsi une plaque d'athérome qui prend un aspect granuleux et jaunâtre et, fait saillie dans la lumière de l'artère. Progressivement, la plaque rétrécit le calibre de vaisseaux, et peut finir par l'obstruer. Certains point de la plaque d'athérome s'ulcèrent. Il s'y dépose des caillots sanguins (thromboses) qui peuvent achever d'oblitérer le tronc artériel.

En réduisant le flux sanguin dans les artères coronaires, la plaque d'athérome diminue de façon plus ou moins sévère et plus ou moins prolongée, l'apport en oxygène du muscle cardiaque. Si cet apport est suffisant au repos, il devient insuffisant à l'effort, au cours duquel le manque d'oxygène se traduit par une brève douleur cardiaque qui cède au repos: c'est l'angine de poitrine. Par contre si le

manque d'oxygène survient de façon intense et durable, par exemple après thrombose d'une grosse artère, il y'a nécrose des cellules cardiaques. Cette nécrose qui constitue le substrat anatomique de l'infarctus s'accompagne de douleurs intenses et prolongées.

Facteurs à risque

La notion de facteurs à risque est née des grandes enquêtes épidémiologiques dont le but étaient de chiffrer, statistiquement en main la fréquence avec laquelle tel facteur se retrouvait dans telle pathologie. C'est à elle que nous devons connaître le rôle joué dans l'infarctus myocardique par:

-l'hérédité
-l'hypertension
-le diabète
-l'angine de poitrine
-le tabac
-l'âge
-le cholestérol
-le HDL cholestérol (High Density Lipoprotéine)
-la triglycéride.

Depuis plus d'une décennie, on sait que la forte pression sanguine est un facteur important de maladie cardio-vasculaire. Une pression accrue dans les artères est préjudiciable tant aux vaisseaux sanguins eux même, notamment à ceux qui alimentent le cerveau et le coeur qu'au muscle cardiaque qui, pour compenser, doit pomper davantage (maladies de coeur par l'hypertension).

Or, on sait que l'hypertension est liée à l'obésité et à certaines réactions affectives aux conditions de vie, surtout dans la société industrielle, réactions qui ne sont pas encore bien comprises. La consommation excessive du sel est également liée à cet état. De fait la réduction du poids et la diminution de la consommation de sel s'accompagnent souvent d'une réduction de la pression sanguine.

Un autre facteur clairement responsable de maladies cardio-vasculaires, l'athérosclérose, provient du dépôt excessif de matières grasses à l'intérieur des artères qui apportent le sang au coeur et au cerveau. Ce phénomène ainsi que les maladies cardio-vasculaires qui en résulte et notamment les maladies coronariennes, est effectivement lié à la consommation de matières grasse. Plus précisément, les pays où l'on consomme plus de graisses dures (appelées graisses saturées) que les graisses plus huileuses (non saturées) ont davantage de maladies coronariennes. La consommation de beurre et de viandes semble en effet contribuer fortement à l'élévation du taux de cholestérol dans le sang, qui à son tour constitue un facteur clé de l'athérosclérose. La relation probable est :

forte consommation de graisses saturées ⇒ fort taux de cholestérol dans le sang ⇒ athérosclérose ⇒ fréquences élevée des maladies coronariennes.

Le cholestérol est transporté dans le sang sous forme de complexes lipidiques et protéiques appelés les lipoprotéines : ces lipoprotéines sont de deux types, les LDL (Low Density Lipoprotéines) et les HDL (High Density Lipoprotéines). Ce sont les LDL qui sont responsables de la fixation de cholestérol sur les parois des vaisseaux, de l'obturation des artères et, par voie de conséquence, des angines de poitrines (violentes douleurs cardiaques), de l'infarctus du myocarde, des thromboses artérielles (obturation des vaisseaux)...

En revanche, les HDL n'interviennent pas pour encrasser les artères, au contraire elles ont un rôle de protection et l'on parle parfois à leur propos de bon cholestérol. Comme elles sont sous la dépendance des hormones femelles, les femmes sont mieux protégées que les hommes des atteintes cardio-vasculaires avant la ménopause.

De même, d'importantes études ont mis en évidence la relation entre la consommation de cigarettes et la fréquence des maladies coronariennes et du cancer du poumon. Bien que la consommation de cigarettes n'augmente le risque de maladies coronariennes que d'environ deux fois, alors qu'elle décuple le risque de cancer des poumons, cette habitude est néanmoins responsable de beaucoup plus de décès par maladies de coeur que par cancer, car les maladies coronariennes, reconnues maintenant comme d'origines

multiples sont beaucoup plus courantes que le cancer causé par les cigarettes.

Outre ces facteurs de risque de maladie cardio-vasculaire, surtout coronarienne (hypertension, cholestérol, angine de poitrine, tabagisme, HDL cholestérol et l'âge), on reconnaît aujourd'hui un autre facteur qui est la perturbation du métabolisme du glucose "diabète" qui se traduit par un taux trop élevé du glucose dans le sang, par des émissions d'urines abondantes et sucrées.

Plusieurs études épidémiologiques, ont essayée de déterminer, comment les facteurs à risques de l'infarctus du myocarde se conjuguent dans la population, pour améliorer la reconnaissance rapide et la prédiction de cette maladie dans des cas individuels et alors, faire les recommandations pour les premières préventions. Parmi ces études on trouve celle faite par l'équipe de G.Assman et H.Shulte et sur laquelle se base ce mémoire.

G.Assman et H.Shulte ont examiné le personnel de divers compagnies et du service publique pour détecter les facteurs à risque de l'infarctus du myocarde (IM) et ont mis sous observation des individus dès l'apparition des signes signifiant d'une attaque.

Un des points fondamentaux était d'étudier l'importance des paramètres des lipides et

des lipoprotéines (molécule résultant de l'union d'une protéine et d'un corps gras) pour la prédiction des maladies coronariennes.

Plus de 18.000 sujets ont participés à cette étude en remplissant un questionnaire concernant:

- leur historique médical et celui de leur famille
- leur consommation de nicotine et d'alcool
- leur activité physique
- leur tension artérielle
- leur poids et leur hauteur.

Aussi on a réalisé un électro-cardiogramme (E.C.G) et un prélèvement de sang sur lequel on a mesuré 20 paramètres de laboratoires et l'intéressé est informé sur les résultats de l'analyse ainsi que son médecin traitant. Un questionnaire est envoyé aux participants tous les deux ans pour établir les nouveaux cas d'IM.

Les deux tiers des participants étaient des hommes et un tiers étaient des femmes qui se répartissent comme suite :

- un tiers avait un âge inférieur à 35 ans
- un tiers avait un âge entre 35 et 45 ans
- un tiers avait un âge compris entre 45 et 65 ans.

468 participants étaient exclus car ils avaient déjà contracter un IM dans le passé.

L'évaluation était limitée sur 6.391 hommes dont l'âge se situe entre 45 et 65 ans, parmi lesquels 5.889 ont retourné le courrier après les deux ans, et seulement 1.674 étaient examinés pendant les quatres années précédentes.

Ainsi, 45 sont morts durant les quatres années dont 14 morts de maladies coronariennes, 4 mort de complication artérielle, 12 morts de cancer, 7 mort d'autres causes et six due au suicide à la violence ou aux accidents.

Parmi les survivants il y'a eu: 31 IM non-fatals et 7 attaques non-fatals. 1.591 survivants n'ont pas contracté d'IM.

On peut retenir de cette étude que l'incidence d'infarctus du myocarde(IM) augmente avec l'âge. On passe d'un risque de 8°/°° pour un âge de quarante ans à 60°/°° pour un âge de 65 ans.

L'incidence des IM dans le groupe des participants qui ont un antécédent familiale double par rapport à ceux qui n'ont en pas (25°/°° à 54°/°°).

L'incidence des IM dans le groupe des participants qui ont souffert d'angine de poitrine

atteind 70°/°° alors que dans le groupe des sujets qui n'ont en pas souffert n'étaient que de 22°/°°.

Les fumeurs courent un risque trois fois plus élevé que les non fumeurs.

Le risque pour les participants présentant une tension artérielle élevée (>160 mm Hg) et deux fois plus grand que ceux qui ont une tension normale (<140 mm Hg).

Les personnes diabétiques courent un risque, deux fois plus élevé que les personnes non diabétiques.

Les personnes présentant un taux de triglycéride entre 150 et 200 mg/dl sont classés dans un groupe à risque assez élevé.

Les individus qui ont un taux de cholestérol total élevé (>260 mg/dl) courent un risque deux fois plus grand. Cependant, on n'a pas pu établir une différence de risque pour les individus qui ont un taux de cholestérol normal (<220 mg/dl) et un peu plus élevé.

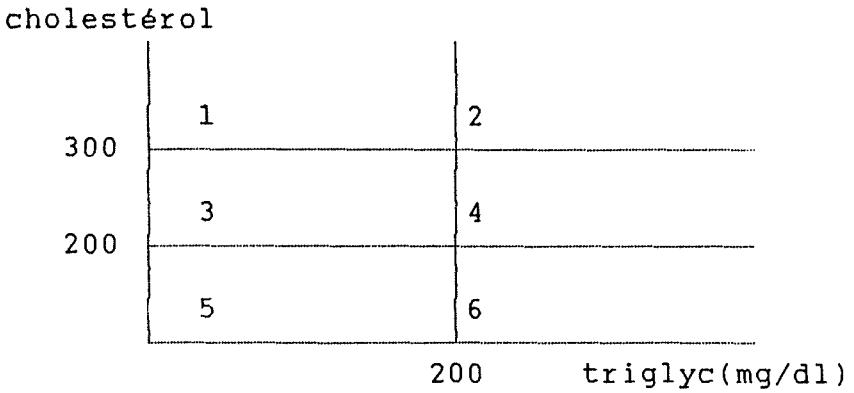
Les sujets avec un taux élevé de LDL (Low Density lipoprotéine) supportent un risque deux

fois plus élevé, cependant pour une valeur intermédiaire le risque ne change pas.

Le meilleur paramètre testé pour établir le risque de contracter un IM est le HDL cholestérol (High Density lipoprotéine). On constate que le risque est 7 fois plus élevé pour une valeur de HDL inférieur à 35 mg/dl que pour une valeur supérieure à 35 mg/dl.(87°/°° contre 13°/°°).

On constate aussi que le rapport chol/HDL nous permet une bonne prédiction du risque; ainsi pour un rapport chol/HDL d'une valeur située entre 5 et 6,5 le risque est de 7°/°°, tandis que pour un rapport supérieur à 6,5 le risque est de 70°/°°.

Les sujets avec une valeur de cholestérol supérieure ou égale à 300 mg/dl sont classés comme des individus à haut risque, et ceux avec les valeurs de cholestérol et de triglycéride inférieurs ou égales à 200 mg/dl sont classés comme des individus à bas ou moyen risque. Les sujets restant sont classés suivant la valeur de HDL (champs 3,4 et 6), si HDL est inférieur à 35 mg/dl ils sont classés comme sujets à haut risque, sinon ils sont classés comme sujets à moyen risque.



Il est connu que plus on accumule de facteurs à risque plus le risque est élevé.

L'utilisation de la méthode de discrimination logistique nous permettera:

- de classer les individus de l'étude dans l'un des groupes des malades ou non

- de calculer les coefficients attachés aux facteurs à risque en maximisant la fonction de vraisemblance et en utilisant la méthode de Newton-Raphson pour résoudre le système d'équations non linéaires qui en découle.

Dès que ces coefficients sont connus la méthode de sélection pas-à-pas nous permet de sélectionner les variables qui interviennent de façon significative,et d'omettre celles qui n'apportent pas de contribution dans le calcul du risque de,faire un infarctus du myocarde. Ce calcul est basé sur le résultat d'une étude cumuléesur une période de quatre

années. Malheureusement, nous n'avons pas pu disposer des données de l'enquête, et dans l'attente, que la clinique saint camille dispose des donnée d'une enquête, nous proposons un programme qui calcule les coefficients des facteurs à risque, ainsi que le risque de contracter un IM.

Et pour conclure, nous proposons un mode d'emploi et une interface agréable au corps médical utilisateur qui lui, permettra de gérer la base de données des individus qui interviennent dans l'étude, de calculer les coefficients des facteurs à risque, et de calculer le risque de faire un infarctus du myocarde.

I.2 Etudes rétrospectives - Etudes prospectives

Dans ce chapitre, nous abordons le rôle et les limites des investigations rétrospectives de facteurs pouvant être associés à l'occurrence d'une maladie. Nous traitons aussi de l'étude prospective en relation avec l'étude rétrospective.

Des associations entre l'occurrence de la maladie et des facteurs peuvent être fausses ou trompeuses. Les mauvaises associations peuvent provenir de groupes de contrôles non appropriés. C'est pourquoi nous allons étudier le choix des contrôles. Il est à remarquer que la possibilité d'associations trompeuses peut être minimisée en contrôlant ou en combinant des facteurs qui produisent de telles associations.

I.2.1 Notions d'études prospectives et d'études rétrospectives

Les enquêtes rétrospectives ont pour but d'étudier les liaisons existant entre un phénomène A, présent au moment de l'enquête et un phénomène B, antérieur, qui est le plus souvent appréhendé par l'interrogatoire. Une étude rétrospective consiste à prendre des échantillons séparés d'individus qui possèdent la maladie sous étude, personnes appelées cas, et à prendre des échantillons d'individus qui n'ont pas cette maladie, personnes appelées contrôles.Le statut de la maladie est regardé dans

cette étude comme une variable fixée.Des variables spécifiant les facteurs de risques sont regardées comme aléatoires et dépendantes du statut de la maladie.

Ce type d'étude pourrait être considéré comme une extension naturelle de la manière de pratiquer des médecins qui souvent partent des histoires des cas pour aider à établir leur diagnostic concernant la maladie.

De même les enquêtes prospectives permettent de suivre un groupe de sujets, afin d'étudier les phénomènes qui les affectent au cours du temps. l'enquête prospective comporte deux phases essentielles:

- une qui correspond au choix de la population et aux premières investigations
-intervient ensuite une phase de surveillance des cas inclus dans l'enquête.

Un échantillon d'individu est pris dans la population qui présente un intérêt et les facteurs de risque sous étude sont pris comme des variables fixées. Nous suivons l'échantillon durant un certain temps pour déterminer l'incidence de la maladie qui est considérée comme un événement aléatoire.

L'étude rétrospective a différents avantages et inconvénients par rapport à l'étude prospective.

1. l'étude rétrospective fournit des résultats sur des données qui sont directement collectables tandis que l'étude prospective requiert habituellement une observation future des individus pendant une période de temps étendue.

2. Pour des maladies de faible incidence, une étude rétrospective donne des échantillons séparés des cas et de contrôles d'une taille d'échantillonnage totale plus petite que celle requise pour une étude prospective car cette dernière conduit à une longue période d'observation .Dans ce type de maladie, l'étude rétrospective peut être la seule approche possible. Pour des maladies plus fréquentes, la disparité entre les tailles requise pour les deux types d'études sera progressivement réduite.

3. L'étude rétrospective est mieux adaptée aux ressources limitées d'un investigateur individuel et elle coûte moins cher que l'étude prospective.
Elle permet de découvrir différents effets jusqu'alors ignorés. Les effets seront étudiés plus en profondeur par d'autres techniques. Il faut cependant s'assurer que l'expérience peut être reproduite par d'autres personnes dans d'autres lieux.

4.Souvent après une étude prospective, nous effectuons une analyse rétrospective sur les mêmes données. Ceci permet de rassembler plus d'informations sur des points découverts au cours de l'approche prospective et elle peut aussi amplifier des

associations apparaissant dans les résultats lors de la première approche.

5.Une étude rétrospective donne ces résultats sous forme d'énoncés sur les associations entre les maladies et les facteurs plutôt que sur les relations de cause à effet. Ceci est dû au fait que cette étude ne sait pas distinguer parmi les formes d'associations possibles (causes à effets, associations dues à des causes communes,...).
Elle rassemble des facteurs associés avec le fait de devenir un sujet malade ou non, plutôt que des facteurs associés à la présence ou non de la maladie.
L'étude rétrospective est plus exposée que l'étude prospective à des associations trompeuses.

I.2.2 Nature représentative des données

la supposition de base pour faire une application pratique avec un échantillon rétrospectif est de s'assurer que les cas et les contrôles rassemblés sont représentatifs de la population spécifiée pour l'investigation.

Ces échantillons doivent être représentatifs dans le sens que la probabilité pour qu'un individu soit échantillonné est indépendante des facteurs de risque que nous étudions. Ceci oblige l'investigateur à mieux examiner les données et le cadre dans lequel elles ont été collectées, de même à regarder si les données ne pourraient pas donner des

facteurs semblant être reliés à une maladie alors qu'il n'y a pas d'association réelle du facteur avec le statut de la maladie.

Souvent les cas et les contrôles pour une étude rétrospective sont pris parmi les patients hospitalisés plutôt que des personnes de la population générale. Ceci facilite les associations trompeuses . Le fait ne concerne pas les personnes hospitalisées mais n'importe quel groupe spécial de personnes utilisé comme source de cas et de contrôles. L'investigateur doit

toujours équilibrer les risques auxquels il est confronté, et décider s'il est important de détecter un effet qui est présent ou de le rejeter quand il ne peut pas refléter la situation véritable.

I.2.3 Les contrôles

Collecter les contrôles, parmi les contrôles hospitaliers, donne lieu à une moins grande dépense et à une plus grande accessibilité.

Cependant ce type de données doit être représentatif de la population plus générale.Il est certain que si les données provenant du milieu hospitalier et de la population générale sont en accord, ceci élimine certaines alternatives dans l'interprétation des découvertes.

Lorsque les deux ensembles de données conduisent à des résultats substantiellement différents, l'investigateur doit se montrer prudent.

Lorsque les contrôles hospitaliers sont choisis, certaines précautions doivent être prises:

1. Quand il y a une preuve que deux maladies sont associées, nous ne pouvons pas utiliser l'une comme contrôle de l'autre à moins que l'étude ne soit conçue pour étudier certains aspects de la relation entre deux maladies.

2. Lorsqu'une disparité entre les cas, les contrôles hospitaliers et les contrôles de la population générale est présente sur plusieurs caractéristiques qui ne sont pas reliés aux hypothèses de l'étude, ceci est un signe avertisseur de la nature non représentative des cas et des contrôles.

3. pour voir si on admet un patient comme cas ou contrôle,nous avons besoin de connaître son histoire.Ceci se fait à l'aide d'une interview.

Il est important qu'une association entre les facteurs de risque et une maladie puisse être répétée avec d'autre données de cas contrôles. Quand une association est très marquée , aucune donnée quantitative supplémentaire n'est requise pour percevoir sa signification.

Lors d'une étude rétrospective, nous aurons peut-être à considérer une association entre un cas et un contrôle ou entre un cas et plusieurs contrôles pour éviter des associations fausses.

Lorsque des facteurs tels que l'âge, la race, le sexe,.. sont connus et fortement suspectés d'être liés à l'occurrence de la maladie, ils doivent être pris en considération et ils donnent lieu à une étude associée de cas-contrôles sur base de ses facteurs.

Enfin, les contrôles sont tels que si une maladie se produit sporadiquement ou son occurrence n'est pas limitée à un groupe bien défini, un choix pour les contrôles n'est pas évident.

CHAPITRE II.ETUDE STATISTIQUE

II 1 INTRODUCTION

Dans une monographie ,Cox développe des méthodes d'analyses purement binaires.Tout au long de son étude,il fait une large comparaison avec l'usage des modèles linéaires appliqués à des données distribuées normalement.

Soit n individus et sur chacun d'eux nous considérons une observation représentée par une variable indicatrice yi où la probabilité de succès est

$$P[yi=1]=\theta i.$$

Le problème auquel Cox s'attaque est de développer des méthodes pour analyser n'importe quelle dépendance de θi en fonction de variables explicatives ou régressives. Il y'a deux approches possibles:

- l'une à l'aide de l'utilisation des tables de contingence

- l'autre en utilisant un modèle de "régression".

C'est à cause de ces développements que Cox a introduit le modèle logistique LINEAIRES.

Les différentes raisons pour introduire le modèle logistique linéaire sont:

- Si nous appliquons la théorie normale des modèles linéaires aux données binaires pour établir une dépendance de $\theta i=P[yi=1]$ en fonction des variables explicatives nous pouvons considérer le modèle

$$(1) \qquad \theta i = E[yi] = \sum_{s=1 \leq s \leq p} a[i,s]*\beta[s]$$

et pour tout $1 \leq i \leq p$

Ceci est l'écriture linéaire par rapport à des paramètres $\beta[s]$

où

$a[i,s]$,sont des constantes connus
$\beta[s]$,sont des paramètres.

Matricièllement nous pouvons écrire:

$$E[yi]=a[i]*\beta= \theta i$$
$$E[Y]=a*\beta =\theta$$

expression de θ en fonction de β

où

θ est un vecteur (1,n) de probabilités avec
$\theta'= [\theta 1,...\theta n]$

Y est un vecteur (1,n) d'indicatrice avec $Y'=[y1,...yn]$

ß est un vecteur (p,1) de paramètres inconnus avec ß'=[ß1...ßn]

a est une matrice de coefficients connus (n,p)

a[i] est la iéme ligne de la matrice a

Nous appliquons la méthode des moindres carrés directement sur les observations binaires dans le but d'estimer ß .Les estimateurs des moindres carrés satisfont aux équations linéaires

$$(2) \qquad (a'*a)*ß = a'*y$$

En effet, on doit trouver ß qui minimise la somme des carrés

c à d que ß est solution de

$$\delta/\delta ß (Y-a*ß)'*(Y-a*ß) = 0$$

d'où nous obtenons (2).

Cette méthode a des limites

1. Var(yi) n'est pas une constante pour tout i. En effet comme $yi^2=yi$ nous avons

$$Var(yi)=E[yi]^2-(E[yi])^2=\theta i-\theta i^2=\theta i*(1-\theta i)$$

or la théorie normale des modèles linéaires requiert que y1,..,yn soient des v.a.i.i d normales avec une variance constante.

Nous pouvons avoir des conditions moins fortes telles que:

$$\text{cov}(y_i,y_j)=0 \text{ pour tout } i,j,i \text{ différent de } j$$
$$\text{Var}(y_i)=\sigma^2 \text{ pour tout } i$$

dans la théorie du second ordre.Dans les deux cas, la condition de la variance constante est requise.
Pour que $\theta i*(1-\theta i)$ soit une constante pour tout i, nous devons avoir les θi tous les mêmes, le cas n'est pas intéressant.

Si les θi varient avec i, il peut y avoir une perte d'information lorsque nous utilisons les estimations des moindres carrés non pondérés.

2.La restriction la plus sérieuse de l'utilisation du modèle (2) est la suivante:

θi représente une probabilité d'où

$$0 \leq \theta i \leq 1 \text{ pour } i=,..,n \quad (3).$$

Il est possible que les estimations des moindres carrés nous conduisent à un vecteur de valeurs ajustées ß où certaines composantes ß[i] sont telles que a[i]*ß=θi ne satisfait pas (3).

Dès lors nous devons considérer des estimations modifiées des moindres carrés obtenues par minimisation par rapport à ß de la somme des carrés (Y-a*ß)'*(Y-a*ß) soumis aux contraintes que tous les éléments a*ß satisfont (3).Ceci donne des résultats difficiles à calculer.

II.2 Formulation du problème

II.2.1 Discrimination Logistique: cas de deux groupes

A l'origine de la méthode, on trouve une relation simple entre les probabilités a posteriori et la fonction de discrimination

Envisageons d'abord le problème de classement d'un individu dans deux groupes π1 et π2 sur la base d'un vecteur d'observation x'=(x1,...,xn).

Soient p1 et p2 les probabilités a priori des groupes π1 et π2 (la probabilité a priori est la probabilité d'avoir la maladie avant d'avoir la connaissance du signe. p1 + p2 = 1) et désignons par f1(x) et f2(x) les densités de x dans chacun des deux groupes.En Application du théorème de Bayes, la probabilité d'appartenir au groupe π2 conditionnelle au vecteur x,appelée aussi probabilité a posteriori de π2, est donnée par la relation

$$Pr(\pi 2|x) = Pr(\pi 2)*Pr(x|\pi 2)/[Pr(\pi 2)*Pr(x|\pi 2)+Pr(\pi 1)*Pr(x|\pi 1)]$$

$$= p2*f2(x)/[p2*f2(x) + p1*f1(x)] \qquad (II.1)$$

$$= 1/[1 + p1*f1(x)/p2*f2(x)]$$

on trouve alors

$$Pr(\pi 2|x) = \frac{1}{1 + \frac{p1}{p2}*\Gamma(x)}$$

avec $\Gamma(x) = f1(x)/f2(x)$.

Bien entendu, $Pr(\pi 1|x) = 1 - Pr(\pi 2|x)$.

La régle de classement optimale c'est à dire celle qui maximalise la probabilité totale de classement correct, consiste à classer l'individu dans le groupe $\pi 2$ si

$$Pr(\pi 2|x) \geq Pr(\pi 1|x)$$

si et seulement si

$$Pr(\pi 2|x) \geq 1 - Pr(\pi 2|x)$$

si et seulement si

$$1/[1 + p1*\Gamma(x)/p2] \geq 1 - 1/[1 + p1*\Gamma(x)/p2]$$

si et seulement si

$$2/[1 + p1*\Gamma(x)/p2] \geq 1$$

si et seulement si

$$2 \geq 1 + p1*\Gamma(x)/p2$$

si et seulement si

$$1 \geq p1 * \Gamma(x)/p2$$

si et seulement si

$$p2/p1 \geq \Gamma(x)$$

si et seulement si

$$\Gamma(x) \leq p2/p1$$

II.2.1.1 Distribution Multinormales

Supposant, que la distribution de x dans $\pi 1$ et $\pi 2$ soit multinormale de moyenne distincte $\mu 1$ différente de $\mu 2$ mais de matrice de variances covariances Σ. Dans ces conditions la probabilité a posteriori de $\pi 2$ devient:

$$Pr(\pi 2|x) = 1/[1 + p1*f1(x)/p2*f2(x)]$$

avec $fi(x) = \exp[-\frac{1}{2}*(x - \mu i)'*\Sigma^{-1}*(x - \mu i)]$; et $i=1,2$.

ce qui donne si on pose $b=p1/p2$

$$1/Pr(\pi 2|x) = 1 + p1*f1(x)/p2*f2(x)$$

$$=1+b*\exp[-\tfrac{1}{2}*(x-\mu 1)'*\Sigma^{-1}*(x-\mu 1)+\tfrac{1}{2}*(x-\mu 2)'*\Sigma^{-1}*(x-\mu 2)]$$

$$=1+b*\exp[(\mu 1-\mu 2)'\Sigma^{-1}*x-\tfrac{1}{2}(\mu 1-\mu 2)'*\Sigma^{-1}*(\mu 1+\mu 2)]$$

$$=1+\exp(\ln(b))*\exp[(\mu 1-\mu 2)'\Sigma^{-1}*x-\tfrac{1}{2}(\mu 1-\mu 2)'*\Sigma^{-1}*(\mu 1+\mu 2)]$$

$$=1+\exp[\ln(b))-\tfrac{1}{2}(\mu 1-\mu 2)'*\Sigma^{-1}*(\mu 1+\mu 2)+(\mu 1-\mu 2)'*\Sigma^{-1}*x$$

$$1/Pr(\pi 2|x) =1+\exp[\alpha 0 + \alpha'*x];$$

d'ou

$$Pr(\pi 2|x) = 1/1+\exp[\alpha 0 + \alpha'*x] \qquad (II.2)$$

où

$$\alpha'=(\mu 1-\mu 2)'*\Sigma-1 \qquad (II.3)$$

et

$$\alpha 0=\ln(p1/p2)-½*\alpha'*(\mu 1+\mu 2)$$

L'expression $\alpha 0 + \alpha'*x$ n'est autre que la fonction linéaire discriminante classique, et il est important de noter que la relation existant entre probabilité et fonction discriminante a la forme logistique. C'est cette relation simple qui est à la base de la méthode de discrimination logistique décrite dans ce chapitre. Notons également que

$$Pr(\pi 1|x) = \exp[\alpha 0 + \alpha'*x]/1+\exp[\alpha 0 + \alpha'*x] \qquad (II.4)$$

II.2.1.2 <u>GENERALISATION</u>

Si Σ désigne une matrice de dispersion d'une distribution multinormle quelconque et $\mu 1, \mu 2$ deux vecteurs de moyennes, il est facile de montrer que la relation logistique (II.1) est conservé pour la famille de distributions suivante:

$$fi(x)=ci*\exp[-½(x-\mu i)'\Sigma-1(x-\mu i)]*h(x) \quad (i=1,2) \quad (II.5)$$

où h(.) est une fonction arbitraire de x intégrable non négatives, et ci une constante telle que fi(.) soit une densité de probabilité. Dans ces conditions, les relations (II.3) s'écrivent

$$\alpha' = (\mu 1 - \mu 2)'\Sigma^{-1}$$

et

$$\alpha 0 = \ln(c1*p1/c2*p2) - \tfrac{1}{2}\alpha'*(\mu 1+\mu 2) \qquad \text{(II.6)}$$

et on constate donc que la fonction h(.) n'intervient pas dans la fonction discriminante. En particulier , si $h(x) \equiv 1$, on retombe sur la loi multinormale classique. De manière moins évidente, si

(i) $\Sigma = I$
(ii) $h(x) = 1$ lorsque $x_i=0$ ou 1, $1 \leq i \leq p$
$h(x) = 0$ sinon,

on retombe sur la loi conjointe de p variables de Bernoulli indépendantes

En supprimant la condition (i), les variables de Bernoulli sont corrélées. Si l'on restreint la condition (ii) à certaines composantes du vecteur x , on obtient un modèle mixte où certaines variables sont continues et d'autres dichotomiques.

Il est bien connu que, dans certaines applications médicale,les variables présentent souvent une distribution normale dans la population des

individus supposés en bonne santé, mais résolument dissymétrique dans la population pathologique. Une telle situation est également couverte par le modèle (II.5). Il suffit de garder la fonction h(x) constante dans la population normale, mais de la laisser croître dans la population pathologique. Ces exemples montres à suffisance la très-grande généralité du modèle logistique. En pratique il n'est pas déraisonnable de postuler d'emblée la forme logistique

$$Pr(\pi 2|x) = 1/1+\exp[\alpha 0 + \alpha' * x]$$

comme base de discrimination de classement,d'autant que la définition de h(.) n'est pas nécessaire. La présence de cette fonction indéterminée rend la méthode robuste.

II.2.2 Cas de k groupes

Anderson a montré que la forme logistique (II.2) se généralise aisément au cas de k groupes $\pi 1,\pi 2,...\pi k$ de probabilité a priori respective p1,p2,...pk et on a

$$Pr(\pi s|x) = \exp[\alpha 0s + \alpha s' * x] * Pr(\pi k|x) \qquad s=1,....,k-1$$

et

$$Pr(\pi k|x) = 1/(1+\sum_{s=1}^{k-1} \exp[\alpha 0s+\alpha s' * x]) \qquad (II.7)$$

Dans la suite, nous nous limiterons au cas de deux groupes, mais il est claire que tous les

résultats et commentaires s'étendent sans difficulté au cas général.

II.3 PROBLEMES D'ESTIMATION

En pratique,le problème se pose d'estimer les coefficients de la fonction logistique discriminante $\alpha 0 + \alpha' * x$.Pour clarifier les choses, il convient de faire une distinction suivant que l'échantillon d'effectif n dont on dispose est extrait du mélange des deux groupes $\pi 1$ et $\pi 2$ ou séparément de chacun des groupes. En effet, si n1 et n2 désignent respectivement l'effectif des échantillons de $\pi 1$ et $\pi 2$, dans le premier cas n1 et n2 sont aléatoires et dans le second ils sont fixés. D'un point de vue statistique cette distinction est essentielle.

II.3.1 Echantillon du mélange

II.3.1.1 Définition de la fonction de vraisemblance

Envisageons d'abord le problème d'estimation lorsque l'utilisateur dispose d'un échantillon de n observations extraites du mélange $\pi = \pi 1 \cup \pi 2$ des deux groupes envisagés. A cet effet, on introduit une variable z telle que

z = 0 si l'individu provient de $\pi 1$
(groupe des malades)

z = 1 si l'individu provient de $\pi 2$

L'échantillon peut alors s'écrire

$$\{(z_i,x_i)\ ,\ i=1,\ldots,n\}$$

où $x_i' = (x_{i1},\ldots,x_{ip})$ le vecteur observation du iéme individu. la vraisemblance de l'échantillon s'écrit

$$L = \prod_{i=1}^{n} g_1(z_i|x_i)*g_2(x_i) \qquad (II.8)$$

où $g_1(.)$ est la densité x_conditionnelle de z
et $g_2(.) = p_1*f_1(.) + p_2*f_2(.)$ la densité marginale de x.

Or, par hypothèse, conditionnellement à x, z est binomiale de proportion

$$Pr(\pi_2|x) = 1/1+\exp[\alpha_0 + \alpha'*x].$$

Dans ces conditions (II.8) peut aussi s'écrire

$$L=\prod_{i=1}^{n}\left\{\frac{\exp[\alpha_0+\alpha'x]}{1+\exp[\alpha_0+\alpha'x]}\right\}^{1-z_i}\left\{\frac{1}{1+\exp[\alpha_0+\alpha'x]}\right\}^{z_i}g_2(x_i)$$

$$L = L_1(\alpha_0,\alpha)*L_2(\theta) \qquad (II.9)$$

où θ désigne le vecteur de tous les paramètres du problème, c'est à dire $\{p_1,\mu_1,\mu_2,\Sigma,h()\}$.

L'expression (II.9) montre qu'il y'a factorisation de la vraisemblance totale en vraisemblance conditionnelle et marginale.

Si la fonction h(.) était connue avec exactitude, on pourrait rechercher les estimateurs du maximum de vraisemblance de $\mu 1, \mu 2$ et Σ, et donc de $\alpha 0$ et α, en utilisant les relations (II.6). Plus généralement, h(.) étant inconnu, il faut maximiser une vraisemblance contenant une fonction inconnue.

En maximisant chaque facteur L1 et L2 séparément, le second facteur est maximum en rendant les éléments g2(xi) égaux et ce quels que soit $\mu 1, \mu 2$ et Σ

Démonstration:

On sait que $g2(xi) = p1*f1(xi) + p2*f(xi)$

$L2(\theta)=\pi_{ni=1} g2(xi)$ si et seulement si

$\ln\{L2(\theta)\}=\Sigma_{ni=1} \ln\{g2(xi)\}$ si et seulement si

$\delta \ln\{L2(\theta)\}/\delta\theta=\Sigma_{ni=1}\{\delta g2(xi)/\delta\theta\}/g2(xi)=0$

si g2(xi) est non nulle alors on intégrant la dernière égalité par rapport à $\delta\theta$ on trouve que g2(xi) est une constante;Ce qui termine la démonstration.

Par contre , le premier facteur $L1(\alpha 0,\alpha)$ fournit directement les estimateurs de maximum de vraisemblance

de $\alpha 0$ et α, c'est à dire des coefficients de la fonction logistique discriminante. En procédant de sorte, la perte d'efficacité n'est réellement importante que si on disposait d'une information sur la fonction h. Pour résoudre le problème,il faut donc maximiser L1 ou son logarithme.

Afin de clarifier l'exposé, nous poserons x0=1 et xsi=$\alpha 0+\alpha x=\alpha 0x0+\alpha 1x1...$ $+\alpha pxp$ (xsi est le produit scalaire des vecteur α et x). x étant le vecteur observé des facteurs à risque associés à la maladie et α le vecteur des coefficients associé à chacun des x.

xsi[i]=$\alpha 0x[i,0]$ + $\alpha 1x[i,1]$ + ...+$\alpha px[i,p]$ est le scalaire correspondant au iéme individu.

$$L1=\overset{\curvearrowleft}{\pi}\overset{\curvearrowright}{n}i=1\{\frac{\exp[\alpha 0+\alpha' x]}{1+\exp[\alpha 0+\alpha' x]}\}1-zi\{\frac{1}{1+\exp[\alpha 0+\alpha' x]}\}zi$$

$$\ln(L1)=\Sigma_{1\leq i\leq n}\{(1-zi)*\ln\frac{\exp(xsi[i])}{1+\exp(xsi[i])}+zi*\ln\frac{1}{1+\exp(xsi[i])}\}(II.10)$$

$$\ln(L1)=\Sigma_{i\in E1}\ln\frac{\exp(xsi[i])}{1+\exp(xsi[i])}+\Sigma_{i\in E2}\ln\frac{1}{1+\exp(xsi[i])}$$

où

$$E1=\{i:zi=0\}$$
$$E2=\{i:zi=1\}$$

$$\ln(L1)=\Sigma_{i\in E1}-\ln(1+\exp(xsi[i]))+xsi[i]-\Sigma_{i\in E2}\ln(1+\exp(xsi[i]))$$

II.3.1.2 RESOLUTION DES EQUATIONS DE MAXIMUM DE VRAISEMBLANCE

Les estimateurs MV s'obtiennent en résolvant les équations du maximum de vraisemblance

$$\frac{\delta \ln(L1)}{\delta \alpha s} = 0$$

$$\frac{\delta \ln(L1)}{\delta \alpha s} = \Sigma_{i\in E1} x[i,s] - \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])} - \Sigma_{i\in E2} \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])}$$

$$\frac{\delta \ln(L1)}{\delta \alpha s} = \Sigma_{i\in E1} x[i,s]\{1-\frac{\exp(xsi[i])}{1+\exp(xsi[i])}\} - \Sigma_{i\in E2} \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])}$$

$$\frac{\delta \ln(L1)}{\delta \alpha s} = \Sigma_{i\in E1} \frac{x[i,s]}{1+\exp(xsi[i])} - \Sigma_{i\in E2} \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])} = 0$$

les équations à résoudre sont alors

$$\Sigma_{i\in E1} \frac{x[i,s]}{1+\exp(xsi[i])} - \Sigma_{i\in E2} \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])} = 0 \quad s=0,.,p \qquad (II.11)$$

avec $xsi[i] = \alpha 0 + \alpha' xi$

Ce système de p+1 équations à p+1 inconnues n'admet pas de solution analytique. Afin de déterminer le point stationnaire $(\alpha 0, \alpha 1, \ldots . \alpha p)$, on recours à la méthode classique de Raphson-Newton définie par le schéma itératif suivant:

$$\alpha t+1 = \alpha t - H\text{-}1(\alpha t)*c(\alpha t) \quad t=0,1,\ldots .p \qquad (II.12)$$

où

αt est l'estimation du vecteur α à la tème itération du processus

$c(\alpha t)$ est le vecteur gradient des dérivées première de ln(L1) en $\alpha = \alpha t$

$H(\alpha t)$ est la matrice hessienne des dérivées secondes $\delta^2 \ln(L1)/\delta\alpha i \delta\alpha j$ en $\alpha = \alpha t$

II.3.1.3 Méthode de Raphson-Newton

Soit le système $f(X) = 0$ qui comporte n équations non linéaires à n inconnues

$$f(X) = 0 \equiv \begin{array}{l} f1(x1,x2,....xn) = 0 \\ f2(x1,x2,....xn) = 0 \\ . \\ . \\ fn(x1,x2,....xn) = 0 \end{array}$$

La résolution de ce système se ramène à la recherche d'une solution α de l'équation vectorielle $f(X)=0$ en considérant un vecteur de Rn

On calcule le jacobien du système qui est défini par:

$$J(X) \equiv \begin{bmatrix} \delta f1/\delta x1 & \delta f1/\delta x2 & \ldots\ldots\ldots & \delta f1/\delta xn \\ \delta f2/\delta x1 & \delta f2/\delta x2 & \ldots\ldots\ldots & \delta f2/\delta xn \\ . & . & & . \\ . & . & & . \\ fn/\delta x1 & \delta fn/\delta x2 & \ldots\ldots\ldots & \delta fn/\delta xn \end{bmatrix}$$

Le système à résoudre est $f(X)=0$. si α est une solution du système, le développement de Taylor de f autour de α est donnée par

$$f(\alpha) = f(Xn) + J(Xn)*[\alpha-Xn] + ...=0$$

et en faisant l'approximation:

$$f(Xn+1)=f(Xn)+J(Xn)*[Xn+1-Xn]=0\longrightarrow J(Xn)*[Xn+1-Xn]$$

$$=-f(Xn)$$

$$\longrightarrow Xn+1 = Xn -J-1(Xn)*f(Xn)$$

pour cela il faut qu'à chaque itération autour de la solution α que le déterminant de J soit non nul

$$Xn+1 = G(Xn) = Xn -J-1(Xn)*f(Xn)$$

D'un point de vu pratique l'algorithme sera utilisé sous la forme

$$J(Xn)*d(Xn) = -f(Xn) \quad \text{avec } d(Xn) = Xn+1-Xn$$

d(Xn) est la correction à apporter à la solution.

On est passé de la résolution d'un système non linéaire à la résolution d'un système linéaire. La résolution de ce système linéaire peut alors se faire sans utiliser J-1(Xn) et elle fourni les corrections d(X0),d(X1),..,d(Xn). Pour se faire on va utiliser la méthode de Crout.

Pour conclure, ici on est parti du système f(X)=0 et le on résoud avec:

$$J(Xn)*d(Xn) = -f(Xn)$$

et en ce qui concerne notre étude statistique, on part d'une dérivée première

$$\delta \ln(L1)/\delta\alpha i = 0$$

et on va le résoudre alors avec

$$H(Xn)*d(Xn) = -\delta \ln(L1)/\delta\alpha i$$

avec $H(Xn) = \delta^2 \ln(L1)/\delta\alpha i \delta\alpha j$ qui est la matrice Hessienne dont il était question dans (II.12).

Puisque

$$\frac{\delta \ln(L1)}{\delta\alpha s} = \Sigma_{i\varepsilon E1} \frac{x[i,s]}{(1+\exp(xsi[i])} - \Sigma_{i\varepsilon E2} \frac{\exp(xsi[i])*x[i,s]}{1+\exp(xsi[i])} \quad s=0,..p$$

alors

$$\frac{\delta^2 \ln(L1)}{\delta\alpha s \delta\alpha r} =$$

$$\Sigma_{i\varepsilon E1} x[i,s]\{-\frac{\exp(xsi[i])x[i,r]}{\{1+\exp(xsi[i])\}^2}\} - \Sigma_{i\varepsilon E2} \frac{\exp(xsi[i])x[i,s]x[i,r]}{\{1+\exp(xsi[i])\}^2}$$

$$\frac{\delta^2 \ln(L1)}{\delta\alpha s \delta\alpha r} = \Sigma_{i\varepsilon E=E1UE2} \frac{-\exp(xsi[i])}{\{1+\exp(xsi[i])\}^2} x[i,r]x[i,s]$$

Remarque:

Dans le calcul des dérivées premières de ln(L1) il est important de distinguer entre les individus des deux groupes E1 et E2, tandis que dans le calcul des dérivées secondes l'expression des deux termes sous les signes de sommation est identique et c'est pour cela qu' on les a groupé sous le même terme

de sommation mais en indiquant que i appartenait à E égale à la réunion des deux ensembles E1 et E2.

II.3.1.4 La décomposition LU selon la méthode de Crout pour la résolution d'un système d'équations linéaires

Supposons que nous voulons écrire une matrice A sous la forme

$$LU=A \qquad (i)$$

où L est une matrice diagonale inférieure (elle possède seulement les éléments situés sur la diagonale et en-dessous) et U est une matrice diagonale supérieure (elle possède seulement les éléments sur la diagonale et au-dessus). Dans le cas d'une matrice 3x3, les équations (i) s'écrivent

$$\begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} = \begin{bmatrix} \alpha 11 & 0.0 & 0.0 \\ \alpha 21 & \alpha 22 & 0.0 \\ \alpha 31 & \alpha 32 & \alpha 33 \end{bmatrix} \begin{bmatrix} \beta 11 & \beta 12 & \beta 13 \\ 0.0 & \beta 22 & \beta 23 \\ 0.0 & 0.0 & \beta 33 \end{bmatrix}$$

en effectuant la multiplication des deux matrices on obtient le système d'équation suivant:

$$a11 = \alpha 11 \beta 11$$
$$a12 = \alpha 11 \beta 12$$
$$a13 = \alpha 11 \beta 13$$

$$a21 = \alpha 21 \beta 11$$
$$a22 = \alpha 21 \beta 12 + \alpha 22 \beta 22$$
$$a23 = \alpha 21 \beta 13 + \alpha 22 \beta 23$$

$$a31 = \alpha 31 \beta 11$$
$$a32 = \alpha 31 \beta 12 + \alpha 32 \beta 22$$
$$a33 = \alpha 31 \beta 13 + \alpha 32 \beta 23 + \alpha 33 \beta 33$$

on peut utiliser la décomposition (i) pour résoudre le système

$$A*x = (LU)*x = L(U*x) = b \qquad (ii)$$

cela est équivalent à résoudre les systèmes

$$L*y = b \quad \longleftrightarrow \quad \begin{bmatrix} \alpha 11 & 0.0 & 0.0 \\ \alpha 21 & \alpha 22 & 0.0 \\ \alpha 31 & \alpha 32 & \alpha 33 \end{bmatrix} \begin{bmatrix} y1 \\ y2 \\ y3 \end{bmatrix} = \begin{bmatrix} b1 \\ b2 \\ b3 \end{bmatrix}$$

et

$$U*x = y \quad \longleftrightarrow \quad \begin{bmatrix} \beta 11 & \beta 12 & \beta 13 \\ 0.0 & \beta 22 & \beta 23 \\ 0.0 & 0.0 & \beta 33 \end{bmatrix} \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} = \begin{bmatrix} y1 \\ y2 \\ y3 \end{bmatrix}$$

quel est l'avantage de passer de la résolution d'un système d'équations à deux systèmes d'équations? L'avantage est que la résolution d'un système triangulaire est évidente. Voyons qu'est ce que cela donne pour le système L*y= b:

y1=b1/α11

y2=(b2-α21)/α22

y3=(b3-α31b1-α32b2)/α33 vu que αii <> 0.0

ceci peut s'écrire

$$y[1]=b[1]/\alpha[1,1] \qquad \text{(iii)}$$
$$y[i]=\{b[i]-\sum_{1\le j\le i-1}\alpha[i,j]y[j]\}*1/\alpha[i,i] \quad \text{avec } i=2,3,\ n$$

de même pour le système U*x = y on trouve

$$x[n]=y[n]/\beta[n,n] \qquad \text{(iv)}$$
$$x[i]=\{y[i]-\sum_{i+1\le j\le n}\beta[i,j]x[j]\}*1/\beta[i,i] \quad \text{avec } i=n-1,n-2,\ 1$$

étant donné la matrice A, comment trouver les matrices L et U? En effet les équations (i) ont toujours tendance à s'écrire comme

$$\alpha i1\beta 1j + ...=aij$$

le nombre de termes qu'il y a dans la somme dépend de qui de i et j est le plus petit. Cependant nous avons trois cas

i<j: $\alpha i1\beta 1j + \alpha i2\beta 2j.+..+\alpha ii\beta ij.=aij$ (V)

i=j $\alpha i1\beta 1j + \alpha i2\beta 2j.+..+\alpha ii\beta jj.=aij$ (VI)

i>j $\alpha i1\beta 1j + \alpha i2\beta 2j.+..+\alpha ij\beta jj.=aij$ (VII)

Le nombre d'équations de ce système est n^2 et le nombre d'inconnues est n^2+n (les éléments diagonaux αii et βjj sont représentés deux fois) alors le nombre d'inconnues est plus grand que le nombre

d'équations on est amené à choisir n inconnues on choisira

$$\alpha ii \equiv 1 \quad i=1...n \qquad \text{(VIII)}.$$

L'algorithme de Crout consiste à résoudre le système de n^2+n inconnues seulement en arrangeant les équations dans un certain ordre. Cet ordre est le suivant:

- mettre les $\alpha ii=1$ $i=1...n$
- pour chaque colonne $j=1...n$

{* Pour $i=1...j$ (pour tous les éléments au-dessus ou sur la diagonale) en utilisant les équations (V),(VI) et (VIII) on calcule les βij suivant la formule

$\beta ij=aij - \Sigma \alpha[i,k]\beta[k,j]$ somme sur k=1 jusqu'à i-1.

* Pour tout $i=j+1..n$ (tout les éléments situé au dessous de la diagonale) en utilisant l'équation (VIII) on calcule les αij suivant la formule

$\alpha ij=(aij-\Sigma\alpha[i,k]\beta[k,j])/\beta jj$ somme sur k=1 jusqu'à j-1

il faut s'assurer qu'on a exécuter ces deux dernières boucles dans une colonne j avant de passer à la colonne suivante }.

Si on travaille manuellement sur ce ces deux dernières formules, on constate que les α et ß se trouvant sur les membres de droite sont calculés au moment où on a besoin d'eux et que les aij sont toujours utilisés une et une seule fois. Cela signifie que la décomposition aura lieu sur la matrice A de départ. Les éléments diagonaux seront calculés par les ßjj. En ce qui concerne le choix des pivots ßjj figurant dans la formule qui calcule les αij Crout a adopté la méthode de pivot partielle (permutation des lignes) et aussi la méthode de pivot implicite comme expliquées dans le chapitre III (algorithme de la fonction Ludcmp).

II.3.1.5 Considérations pratiques

Il convient de faire un certain nombre de remarques pratiques concernant la résolution des équations MV.

(i)On constate qu'au cours du processus itératif la matrice H-1 varie très lentement en comparaison du vecteur gradient c. Il n'est donc pas nécessaire de la calculer à chaque itération, ce qui, dans certains cas, permet de réaliser un gain de temps appréciable.

(ii) Dans tout processus itératif, il convient de choisir un point de départ approprié. On pourrait, par exemple, se servir des relations (II.3) et remplacer les paramètres $\mu 1, \mu 2, \Sigma, p1$ et $p2 = 1-p1$ par leurs estimations classiques moyenne de x1, moyenne de

x2, S, n1/n et n2/n. Toutefois, l'expérience montre qu'en partant de l'origine, c'est à dire $\alpha t = (0,0,\ldots,0)$, il est rare que l'on ne converge pas après dix itérations avec une précision relative de 10-6.

(iii) Sous certaines conditions de régularité, il est connu que les estimateurs MV sont asymptotiquement convergents, normaux et efficaces. Par conséquent à la dernière étape du processus (au seuil de précision fixé), la matrice {-H-1} constitue une estimation de la matrice de variances-covariances asymptotique des estimateurs α,

$$\text{Vas}(\alpha) = \{E(\frac{\delta \ln(L1)}{\delta \alpha})(\frac{\delta \ln(L1)}{\delta \alpha})'\}-1$$

$$= -\{EH-1\}$$

où l'espérance mathématique E est prise sur l'espace des échantillons.

(iv) Dans certains cas, qui commencent à être connus, le processus itératif diverge, mais nous y reviendrons au paragraphe (II.3.3).

II.3.2 Echantillons séparés

Supposons,à présent, que l'on dispose d'échantillons extraits séparément de chacun des deux groupes $\pi 1$ et $\pi 2$. Dans ce cas les effectifs n1 et n2 sont fixés et n'apportent aucune information quant à la proportion relative de chacun des groupes dans le

mélange. cette situation est la plus fréquemment rencontrée en pratique.

Il est alors indispensable de définir soi même les probabilités a priori p1 et p2, soit qu'elles sont connues, soit qu'elles sont estimées par un autre échantillon. Dans le cas d'échantillons séparés, il n'est plus question d'écrire la relation (II.8) et on affaire à un problème de maximisation avec contraintes sur certains paramètres.

Voyons tout d'abord, s'il existe une relation entre les fonctions logistiques discriminantes obtenues à partir de deux ensembles de probabilité a priori (p1,p2) et (p1*,p2*).

Les densités f1(x) et f2(x) ne dépendant pas des choix des probabilités a priori, on peut écrire

$$fi(x) = \frac{Pr(\pi i|x)g2(x)}{pi} = \frac{Pr^*(\pi i|x)g2^*(x)}{pi^*}$$

En faisant le rapport f1(x)/f2(x), on obtient

$$\frac{f1(x)}{f2(x)} = \frac{Pr(\pi 1|x)g2(x)p2}{Pr(\pi 2|x)g2(x)p1} = \frac{Pr^*(\pi 1|x)g2^*(x)p2^*}{Pr^*(\pi 2|x)g2^*(x)p1^*}$$

$$\longleftrightarrow \quad \frac{Pr^*(\pi 1|x)}{Pr^*(\pi 2|x)} = \frac{Pr(\pi 1|x)p2p1^*}{Pr(\pi 2|x)p1p2^*} \qquad (II.13)$$

$$= \frac{\exp(\alpha 0+\alpha' x)}{1+\exp(\alpha 0+\alpha' x)} * (1+\exp(\alpha 0+\alpha' x)) * \frac{p2p1^*}{p1p2^*}$$

$$= \exp(\ln \frac{p2p1^*}{p1p2^*} + \alpha 0+\alpha' x)$$

$$\frac{Pr^*(\pi 1|x)}{Pr^*(\pi 2|x)} = \exp(\beta 0+ \alpha 0+\alpha' x)$$

où

$$\beta 0 = \ln \frac{p2p1^*}{p1p2^*}$$

Et puisque $Pr^*(\pi 2|x) = 1-Pr^*(\pi 1|x)$
$= 1- \exp(\beta 0+ \alpha 0+\alpha' x)Pr^*(\pi 2|x)$

$\longrightarrow Pr^*(\pi 2|x)*\{1+\exp(\beta 0+ \alpha 0+\alpha' x)\}=1$

ALors

$$Pr^*(\pi 2|x)= \frac{1}{1+\exp(\beta 0+ \alpha 0+\alpha' x)}$$

$$\longrightarrow Pr^*(\pi 2|x)= \frac{1}{1+\exp(\alpha^* 0+\alpha^{*\prime} x)} \qquad (II.14)$$

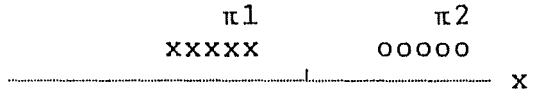
avec $\alpha^* = \alpha$
$\alpha^* 0=\beta 0+ \alpha 0$

Ceci montre qu'il est simple, de passer d'un système de probabilités a priori à l'autre. C'est à partir de cette constatation que J.A Anderson a pu démontrer, du moins dans le cas des variables discrètes, que pour obtenir les estimateurs de MV des coefficients α à partir d'échantillons séparés, il suffit d'utiliser la méthode décrite dans le paragraphe II.3.1 comme si l'échantillon était extrait du mélange et de faire ensuite l'ajustement du terme indépendant $\alpha 0$ comme décrit en (II.14).

Les résultats s'étendent également au cas des variables continues, mais avec une légère perte d'efficacité. Jusqu'à présent,ce problème n'ayant pas reçu de solution définitive, la méthode décrite ci dessus est recommandée jusqu'à ce que de meilleures soient disponibles.

II.3.3 Séparation complète de l'échantillon

Nous avons souligné, au paragraphe II.3.1 que dans certaines situations, le processus itératif diverge. C'est le cas comme nous allons le voir, lorsqu'il y a séparation complète de l'échantillon. Afin de mieux saisir cette notion, on représente les n vecteurs observation xi comme n points de l'espace à p dimensions. de ces n points, n1 appartiennent à $\pi 1$ et n2 à $\pi 2$. On dit qu'il y a séparation complète de l'échantillon lorsqu'il existe un hyperplan qui sépare complètement les deux sous échantillons d'effectif respectif n1 et n2. Ceci est illustré à une dimension dans la figure ci-dessous



Mathématiquement parlant, il existe un vecteur αc tel que

$$\begin{aligned} xsici &= \alpha c'xi > 0 \quad , i \varepsilon E1 \\ xsici &= \alpha c'xi < 0 \quad , i \varepsilon E2. \end{aligned} \qquad \text{(II.15)}$$

Pour ce vecteur, la vraisemblance (II.10) s'écrit

$$\ln(L1(\alpha c)) = \sum_{1\leq i\leq n} \ln \frac{\exp(|xsici|)}{1+\exp(|xsici|)} \qquad (II.16)$$

Si on multiplie αc, par une constante k positive,(II.16) devient

$$\ln(L1(k\alpha c)) = \sum_{1\leq i\leq n} \ln \frac{\exp(k|xsici|)}{1+\exp(k|xsici|)}$$

et donc

$$\lim_{k\to\infty} \ln(L1(k\alpha c)) = 0$$

En d'autre terme la variance atteint son maximum absolu (=1) sur la frontière de l'espace des paramètres. On conclut que, dans le cas de séparation complète, les estimateurs MV sont infinis.

En pratique,nous allons montré que s'il y a séparation complète, tout processus itératif convergent doit nécessairement nous conduire à un vecteur α satisfaisant aux relations (II.15). En effet, la vraisemblance conditionnelle s'écrit

$$L1(\alpha) = \prod_{i\in E1} \frac{\exp(\alpha' xi)}{1+\exp(\alpha' xi)} \prod_{i\in E2} \frac{1}{1+\exp(\alpha' xi)} \qquad (II.17)$$

Supposons qu'à l'étape t du processus itératif il existe un vecteur observation xj, $j\in E2$, qui

soit mal classé. Ceci signifie que, pour cette observation xj,

$$Pr(\pi 2 | xj) = \frac{1}{1+\exp(\alpha' xi)} < \frac{1}{2}$$

puisqu'elle est classée dans π1. Par conséquent, à l'étape t, c'est à dire α=αt (II.17) devient

(II.18)

$$L1(\alpha) = \pi_{i\in E1} \frac{\exp(\alpha t' xi)}{1+\exp(\alpha t' xi)} \; \pi_{\substack{i\in E2 \\ i<>j}} \frac{1}{1+\exp(\alpha t' xi)} \; \frac{1}{1+\exp(\alpha t' xj)} < \frac{1}{2}$$

L'inégalité (II.18) subsiste à chaque itération tant qu'il y a au moins une observation mal classée. Or, puisqu'il y a séparation complète, on sait que le maximum de la vraisemblance (II.17) vaut 1 est atteint sur la frontière. Donc, si le processus itératif converge, il faut bien qu'a une certaine étape ts>t on ait

$$L1(\alpha ts) > 1/2 \quad \text{(ts étape de séparation)}$$

Autrement dit, le vecteur αts "sépare complètement" les deux sous-échantillons. Si donc, à chaque itération du processus, on calcule L1 ou si on comptabilise le nombre total r d'individus mal classés, il suffit d'arrêter le processus itératif dès que

$L1 > 1/2$ ou $r=0$

Bien sûr, dans ce cas , les estimations α obtenues sont arbitraires, mais tous les individus de l'échantillon sont bien classés.

II.4 Méthode de sélection de variables pas-à-pas

Le problème d'estimation de la fonction logistique étant résolu, il est important de proposer une méthode de sélection de variables de manière à n'inclure dans la fonction discriminante que les variables utiles.

La méthode qui suit est une méthode de sélection pas à pas basée sur l'épreuve du rapport des vraisemblances.

Désignons par $L1(\alpha 0,\alpha j)$ la vraisemblance conditionnelle de l'échantillon lorsque seule la variable xj est prise en considération, soit

$$L1(\alpha 0,\alpha j)= \prod_{i\in E1} \frac{\exp(\alpha 0+\alpha j x_{ij})}{1+\exp(\alpha 0+\alpha j x_{ij})} \prod_{i\in E2} \frac{1}{1+\exp(\alpha 0+\alpha j x_{ij})}$$

(i) A la première étape du processus, on sélectionne la variable xi1 pour laquelle l'expression

$$\sup_{\alpha 0,\alpha j} L1(\alpha 0,\alpha j)$$

est maximale. Si L1(1) désigne ce maximum, on peut écrire

$$L1(1) = \max_{1\leq j\leq p} \sup_{\alpha 0,\alpha j} L1(\alpha 0,\alpha j) = L1(\alpha 0,\alpha i1).$$

(ii) A la seconde étape, on choisit parmi les p-1 variables restantes, la variable xi2 qui, conjointement avec xi1 , fournit la vraisemblance maximale L1(2) ,soit

$$L1(2) = \max_{\substack{1\leq j\leq p \\ j<>i1}} \sup_{\alpha 0,\alpha i1,\alpha j} L1(\alpha 0,\alpha i1,\alpha j)$$

$$= L1(\alpha 0,\alpha i1,\alpha i2).$$

(iii) A la kème étape du processus de sélection, le sous ensemble {xi1,xi2,...,xik} est tel que

$$L1(k) = \max_{\substack{1\leq j\leq p \\ j<>i1..ik-1}} \sup_{\alpha 0,\alpha i1,..\alpha j} \{L1(\alpha 0,\alpha i1,..,\alpha ik-1,\alpha j)\}$$

$$= L1(\alpha 0,\alpha i1,..,\alpha ik).$$

(iv) Critère d'arrêt.

On continue ainsi de suite jusqu'à ce que la vraisemblance obtenue L1(k) n'augmente plus de façon significative. Le critère d'arrêt.adopté repose sur le

logarithme du rapport des vraisemblances généralement utilisé dans les épreuves d'hypothèses. En effet, le critère

$$r(k) = 2\ln \frac{L1(k)}{L1(k-1)}$$

est asymptotiquement distribué comme X^2 à 1 degré de liberté. Le processus de sélection s'arrête donc à la (k-1)ème étape si

$$r(k) \leq Qcc(1-\alpha,1)$$

où α est le niveau d'incertitude fixé par l'utilisateur et $Qcc(1-\alpha,1)$ le $(1-\alpha)$-quantile de la distribution X^2 à 1 degré de liberté.

II.5 Remarques finales

L'utilisation de la méthode logistique discriminante est recommandée à de nombreux égards. Tout d'abord, elle s'applique quel que soit le type de distribution envisagé, pour autant que le rapport des logarithmes des densités soit une fonction linéaire des observations. Cette condition est notamment satisfaite pour les distributions multinormles tronquées ou non, mais aussi pour tout une série de distributions conjointes de variables discrètes et continues.

La méthode logistique conduit à une régle de classement simple, puisque linéaire en les variables et par définition même de la méthode, il est aisé de calculer les probabilités a posteriori. Contrairement à beaucoup de méthodes, on estime directement les coefficients de la fonction discriminante sans passer par les paramètres de la population. Cette particularité confère à la méthode logistique robustesse.

Lorsqu'il y a séparation complète de l'échantillon, la méthode doit nécessairement nous conduire à une régle qui classe correctement tous les individus de l'échantillon, même si, dans ce cas, les coefficients estimés sont relativement arbitraires.

Une méthode de sélection pas à pas basé sur le critère du rapport de vraisemblance permet à l'utilisateur d'éliminer les variables redondantes ou n'apportant que peu d'information.

En conclusion, le modèle logistique est général dans la mesure où il est applicable quel que soit le type de variable utilisées, et simple parcequ'il conduit à des fonctions discriminantes en les variables

CHAPITRE III. CONCEPTION

A) ARCHITECTURE LOGIQUE

Cette architecture logique fait intervenir la relation "utilise" afin de faciliter la factorisation du travail et le contrôle de la redondance. Un composant d'un certain niveau ne peut utiliser un composant du même niveau, d'autre part un composant utilise un autre si et seulement si la validité du premier dépend de la disponibilité d'une version correcte du second.

B) DESCRIPTION SUCCINCTE DES MODULES

1 Le module coordinateur : JAWAD

Ce module permet à l'utilisateur de faire un choix de traitement. Parmi ces choix on trouve:

- La création d'un fichier ascii contenant les différents facteurs à risque de chaque individu nécessaire pour calculer les coefficients de la fonction logistique.

- Le calcul des coefficients des facteurs à risque

- Le calcul du risque

- La gestion de la base de donnée

2.Le module Statis

Ce module calcule les coefficients des facteurs à risque.

3.Le module Risque

Ce module calcule le risque de faire un infarctus du myocarde.

4.Le module Entrées

Ce module s'occupe de la saisie des données d'un individu à savoir: nom-prénom, adresse, hérédité, diabète, fumeur, angine de poitrine, âge, la tenssion artèrielle, cholestérol, HDL cholestérol, triglicéride.

5.Le module Sortie

Ce module permet de montrer à l'utilisateur des résultats correctement formatés à l'écran.

6.Le module Accès Base de données

Ce module a pour objectif de fournir aux modules supérieurs les fonctions d'accès à la base de données

C) Spécification externe des modules

1.Module coordinateur JAWAD

Argument :aucun.
Précondition :le choix correspond à une fonction ou à un module.
Résultat :aucun.
Postcondition:la fonction ou le module est appelé.

1.a Fonction Fichier

Cette fonction permet de créer un fichier ASCII à partir de la base de données Patient dont les champs permettent de calculer les coefficients des différents facteurs à risque

Argument : aucun
Précondition :la base de données est dans un état cohérent et les champs la constituant sont syntaxiquement corrects
Résultat :un fichier ASCII est créé.
Postcondition:la base de données reste cohérente.

1.b Fonction Mise à jour

Cette fonction permet de réaliser différentes tâches de mise à jour de la base de données, à savoir:

- La saisie d'enregistrements.
- La modification des champs d'un enregistrement.
- Elimination d'un enregistrement.

Argument : aucun
Précondition : la base de données est dans un état cohérent.
Résultat : - la saisie est bien faite.
- la modification des champs a eu lieu.
- l'enregistrement est éliminé.
Postcondition:la base de donnée reste cohérente.

1.b.1 La fonction Matlook

Cette fonction affiche un écran de saisie d'enregistrement et remplit les différents champs par les champs du premier enregistrement de la base de données Patient.

Argument :aucun.
Précondition :la base de données Patient est ouverte.
Résultat :affiche à l'écran les champs du premier enregistrement.
Postcondition:la base de donnée est non modifiée.

1.b.2 La fonction Matscr.

Cette fonction affiche un écran de saisie et attend l'entrée des valeurs des champs.

Argument :aucun.
Précondition :la base de donnée Patient est ouverte.
Résultat :lit les différents champs par l'intermédiaire d'un masque de saisie
Postcondition:les variables de saisie sont chargées;

1.b.3 Encodage

Argument :aucun.
Précondition :la base de données est ouverte.
Résultat :affichage de l'enregistrement lu à l'écran.
Postcondition:l'enregistrement est encodé.

1.b.4 Modification

Argument :aucun.
précondition :la base de données est ouverte.
Résultat :affichage de l'enregistrement modifié à l'écran.
Postcondition:l'enregistrement est modifié.

1.b.5 Suppression

Argument :aucun.
Précondition :le pointeur est positionné sur l'enregistrement à supprimer.
Résultat :affichage de l'enregistrement précédent.
Postcondition:l'enregistrement est supprimé après confirmation.

1.b.6 Positionnement suivant

Cette fonction affiche l'enregistrement suivant.

Argument :aucun.
Précondition :le pointeur n'est pas positionné sur la fin du fichier.
Résultat :affichage de l'enregistrement suivant.
Postcondition:le pointeur est positionné sur l'enregistrement suivant.

.m.:::1.b.7 positionnement précédent

Cette fonction permet de se positionner sur l'enregistrement précédent.

Argument :aucun.
Précondition :le pointeur n'est positionné sur le début du fichier.
Résultat :affichage de l'enregistrement précédent.
Postcondition:le pointeur est positionné sur l'enregistrement précédent.

1.b.8 Escape

Cette fonction nous fait sortir de la mise à jour.

Argument :aucun.
Précondition :aucune des fonctions de 1.b.1 à 1.b.7 n'est en fonction.
Résultat :effacement des lignes de 16. à 24 et retour au menu principal.
Postcondition:retour au menu principal.

2 Module Statis

Ce module calcule les coefficients des facteurs à risque.

Argument :aucun.
Précondition :les fonctions "Lect", "Mnewt", "Userfun", "Ludcmp", "Lubksb" et écriture accomplissent bien leur tâches.
Résultat :le fichier RESULTAT.DAT est crée.
postcondition:les coefficients des facteurs à risque sont calculés.

2.1 Lect

Cette fonction lit le fichier où sont stockées les données issues de la base de données Patient.dbf, transforme ces données en réel en double précision et les stocke dans une matrice **mat1.

Argument :le nom du fichier qui contient les données des patients.
Précondition :le nom du fichier qui contient les données des patients.existe.
Résultat :une matrice **mat1 est créée.
Postcondition:chaque ligne de la matrice **mat1 contient les données d'un patient.

2.2 Mnewt

Cette fonction résoud un système d'équations non linéaires de n équations à n inconnues.

Argument :- Ntr qui est le nombre d'itération.
- x est un tableau de dimension n qui sera initialisé à une certaine valeur et qui contiendra à la sortie la solution du système.
- n est la dimension du système.
- tolx est la tolérance de x.
- tolf est la tolérance de fi(où fi est une des équations du système non linéaire).

Précondition :la fonction Mnewt a besoin d'une version correcte de la fonction "Lect".

Résultat :le tableau x est garni par les coefficients des facteurs à risque.
Postcondition:x représente la solution du système si celui-ci converge.

2.2.1 Userfun

- cette fonction réalise l'affectation d'une personne au groupe des malades (z[i]=0) ou au groupe des non-malades (z[i]=1)

- elle calcule les bet[j] donnés par:

$$bet[j] = \delta \ln(L)/\delta xj$$

qui est la dérivée première du logarithme népérien de la fonction de vraisemblance L par rapport à xj.

- elle calcule les alpha[i,j] donnés par:

$$alpha[i,j] = \delta^2 \ln(L)/\delta xi \delta xj$$

qui la dérivée seconde du logarithme népérien de la fonction de vraisemblance L par rapport à xi et xj.

Argument :- x tableau qui contient la valeur initiale et la valeur finale solution du système et contient les coefficients des facteurs à risque.
- bet[j].
- alpha[i,j].

Précondition :la fonction "Userfun" a besoin d'une version correcte de la fonction "Lect".

Résultat :aucun
Postcondition:bet[j] et alpha[i,j] sont calculés.

2.2.2 Ludcmp

Cette fonction effectue la LU décomposition (méthode de résolution d'un système d'équations linéaires en décomposant la matrice des coefficients en une matrice triangulaire inférieure (L) et une matrice triangulaire supérieure (U)).

Argument :- **a matrice hessienne des dérivées secondes.
- n qui est la dimension du système.
- *indx qui est un vecteur d'entiers qui contiendra les permutations effectuées.
- *d =±1,dépend du fait si les lignes ont été permutées ou non

Précondition :**a est la matrice hessienne et *bet sont calculées par la fonction "Userfun".

Résultat :**a est transformée en une matrice LU et le vecteur *indx mémorise les permutations qui ont eu lieu au cours de cette décomposition.

Postcondition:**a et *bet contiennent les résultats de la décomposition.

2.2.3 La fonction Lubksb

La fonction "Ludcmp" est utilisée en combinaison avec la fonction "Lubksb" pour résoudre un

système d'équations linéaires.Cette dernière calcule les résultats par substitution.

Argument :- la matrice **a est ici non pas la matrice hessienne mais la matrice issue de la décomposition LU retournée par la fonction "Ludcmp".
- n est la dimension du système .
- *indx est le vecteur permutation retourné par la fonction "Ludcmp".
- *b est le vecteur second membre du système à résoudre et contient les dérivées partielles δln(L)/δxj; ce dernier a été retourné par la fonction "Userfun".

Précondition :la matrice **a est la matrice qui contient la décomposition LU et *indx est le vecteur de permutation qui sont issues de la fonction Ludcmp.

Résultat :*b.

Postcondition:*b contient le vecteur solution du système.

3.Module Risque

Ce module calcule le risque de faire un infarctus du myocarde

Argument :aucun.

Précondition :- les coefficients des différents facteurs à risques sont calculés par le module "Statis" et sont stockés dans un

tableau.
- les valeurs des différents facteurs à risque sont saisies et sont mémorisées dans un autre tableau.

Résultat :le risque de faire un infarctus est calculé et est stocké dans le fichier FICHRISQUE.DAT.
le risque de faire un infarctus est donné par la probabilité a posteriori:
$Pr(\pi 1|\alpha i) = 1-1/(1+\exp(x0 + x'*\alpha i))$
$=1 - 1/(1 + \exp(xsi[i])$

Postcondition:$0 \leq risque \leq 1$.

3.1 Patdon

Cette fonction charge dans un tableau les données d'un patient qui se trouve dans le fichier DONPAT.DAT.

Argument :- nom du fichier où sont stockées les données du patient pour lequel on veut calculer le risque.
- tableau qui contiendra les valeurs des facteurs à risques.

Précondition :les facteurs à risque sont stockés dans le fichier DONPAT.DAT.

Résultat :un tableau est créé.

Postcondition:le tableau contient les différents facteurs à risque.

3.2 Coef

Cette fonction charge dans un tableau les coefficients des facteurs à risque calculés par "Statis" et déposés par ce dernier dans le fichier RESULTAT.DAT.

Argument :- nom du fichier où sont stockés les différents coefficients des facteurs à risque.
- tableau qui contiendra les valeurs des coefficients des facteurs à risques.

Précondition :les coefficients des facteurs à risques sont stockés dans le fichierRESULTAT.DAT.

Résultat :un tableau est crée.

Postcondition:le tableau contient les coefficients des différents facteurs à risque.

3.3 Ecrisq

Cette fonction écrit dans le fichier FICHRISQ.DAT le risque r calculé par le module risque.

Argument :- nom ou sera stocké le risque r.
- une variable qui contiendra la valeur du risque r.

Précondition :les fonctions Patdon et Coef doivent bien accomplir leurs tâches pour que leurs tableaux soient bien garnis.

Résultat :création du fichier FICHRISQ.DAT.

Postcondition:le risque est stocké dans le fichier FICHRISQ.DAT.

2.3 Ecriture

Cette fonction écrit dans le fichier RESULTAT.DAT les valeurs du vecteur solution du système d'équations non linéaires, c.à.d les coefficients des facteurs à risque

Argument :- *fic qui est une chaîne de caractères contenant le nom du fichier où seront déposés les coefficients.
- x vecteur contenant les coefficients des facteurs à risques solution du système d'équations non linéaire.

Précondition :le vecteur x contient les coefficients des facteurs à risques issus de la fonction Mnewt.

Résultat :le fichier RESULTAT.DAT est crée.

Postcondition:le fichier RESULTAT.DAT est crée.

D) ALGORITHMES

1.Algorithme du module "Statis"

1.a Les variables globales

ntrial :est le nombre d'itérations maximum,cet entier doit être.élevé pour mieux approcher la solution si cette solution existe ($100 \leq ntrial \leq 200$).

tolx :représente la tolérance de x au départ cette valeur ne doit pas être élevée (10e-6) pour avoir une idée de l'évolution de la convergence du système.

tolf :tolérance de f vecteur second membre du système.

n :la dimension du système.

x :vecteur de réels en double précision qui contiendra les valeurs de départ et les valeurs finales des coefficients des facteurs de risque.

**mat1 :chaque ligne de cette matrice contiendra les facteurs à risque d'un patient donné. Les dimensions de cette matrice sont a[1..np,1..n].

*param_file:chaîne de caractères qui contiendra le nom du fichier-source "PATIENT.DAT"qui contiendra les données des np patients de la base de données.

*res :chaîne de caractères qui contiendra le nom du fichier destination "RESULTAT.DAT" où seront déposés les coefficients recherchés.

1.b Algorithme du module Statis

-Lecture du fichier PATIENT.DAT par la fonction Lect.

-Résolution du système non linéaire de n équations à n inconnues par la fonction Mnewt.

-Ecriture dans le fichier RESULTAT.DAT des coefficients des facteurs à risque par la fonction Ecriture.

1.1 Algorithme de la fonction Lec(*par_file)

1.1.a variables internes

mat2[1500,10]:matrice qui peut contenir les données de 1500 patients.

ligne[100] :vecteur de 100 caractères qui sera utilisé pour la lecture des données d'un patient.

pos[] :contient la position du premier caractère de chaque champs du vecteur ligne[100].

len[] :contient les longueurs de ces derniers champs.

1.1.b Algorithme de la fonction Lect

-lire la première ligne du fichier "PATIENT.DAT".

-un compteur i comptabilisera le nombre de patients de la base de données.

```
Tant que  not EOF
          {Retrait du caractère EOF du vecteur ligne lu
           et le remplacer par le caractère fin de
           chaîne de caractère '\0'.

           Pour j=1 jusqu' à n-1
               {copiez champs dans  une variable
                transformez cette variable en un nombre
                réel et mettez le résultat dans
```

```
        mat2[i,j]
      }
```

lire la ligne suivante
incrémentez le compteur i

}/*fin de tant que*/

- np=i.

- Mettre dans la première colonne de la matrice mat2 le nombre 1 car dans la formule $P(\pi 1/\alpha)=1/1+\exp(x0*1 + X*\alpha)$ il y a le coefficient x0 qui n'est multiplié par aucun facteur à risque αi.

-Allouez de la place mémoire pour la matrice **mat1.

-Transcrire la matrice mat2 dans la matrice **mat1.

1.2 Algorithme de la fonction Userfun(x,alpha,bet)

1.2.a les variables reçues

x[] :vecteurs des coefficients des facteurs à risques.

**alpha:la matrice hessienne contenant les dérivées secondes du logarithme de la fonction de vraisemblance L.

*bet :vecteur de dimension n contenant les dérivées premières du logarithme de la fonction de vraisemblance L.

1.2.b Les variables internes

*z :vecteur d'entiers qui permet de classer un individu dans l'un des groupes π1 ou π2.

*xsi :vecteur résultat du produit de la matrice **mat1 et du vecteur x.

*gam1,*gam2:vecteurs intermédiaires qui permettent de séparer les calculs concernant les individus des groupes π1 et π2.

1.2.c Algorithme de la fonction Userfun

-On commence par allouer de la place mémoire pour les vecteurs gam1,gam2,*xsi et *z.

- On initialise ces dernières .

- On calcule pour chaque individu i (i=1 jusqu'à np nombre de patients)

$$xsi[i] = \sum_{1 \le j \le n} mat1[i,j]*x[j]$$

- On calcule z[i] :

pour i=1 jusqu'à np

{ si 1/(1+exp(xsi[i]) ≥ exp(xsi[i])/(1+exp(xsi[i]))
alors z[i]=1
sinon z[i]=0
}

- On calcule les

$$bet[j] = -\delta \ln(L)/\delta xj;$$

comme vu au chapitre I, le calcul de la fonction de vraisemblance L distingue entre les individus des deux groupes. Ceci donne alors

pour j = 1 jusqu'à n

```
{ pour i = 1 jusqu'à np
     { si l'individu appartient au groupe π2 (z[i]=1
       groupe des non-malades )

         gam1[j] = gam1[j]+ mat[i,j]/(1+exp(xsi[i]))

       sinon
         gam2[j]=gam2[j]+mat1[i,j] *exp(xsi[i])
                                  (1+exp(xsi[i]))
     }

   bet[j] = gam2[j] -gam1[j]

 }
```

- on calcule la matrice hessienne

$$alpha[j,k] = \delta^2 \ln(L)/\delta x j \delta x k = \Sigma_{1 \le i \le np} - \frac{exp(xsi[i]) * mat1[i,j] * mat1[i,k]}{(1+exp(xsi[i]))^2}$$

- On désalloue les vecteurs *gam1,*gam2,*xsi et *z.

1.3 Algorithme de la fonction Ludcmp(a,n,indx,d)

1.3.a Les variables reçues

**a : matrice carré hessienne.

n : la dimension du système

*indx : vecteur mémorisant les permutation effectuées lors de la décomposition Pa=LU.

*d : quand on interchange deux lignes on change la parité de (*d)=±1.

1.3.b les variables internes

imax:indice de la ligne contenant le plus grand pivot.

*vv :vecteur qui contient la valeur absolue de l'inverse du plus grand élément de chaque ligne de la matrice **a.

big :est égale à l'inverse de vv[i].

et d'autres variables de travail.

1.3.c Algorithme

On remarque que les éléments ßij d'une matrice situés au-dessus de la diagonale ont un indice i<j,que les éléments situés sur la diagonale ont un indice i=j et que les éléments αij situés en-dessous de la diagonale ont un indice i>j.

1/-On commence d'abord par trouver le plus grand élément en valeur absolue de chaque ligne de la matrice **a.

-Si pour une ligne ce big est nul alors la matrice est singulière.

-vv[i]=1/big.

2/-On travaille colonne par colonne.

Pour-chaque colonne k

$$ßij=a[i,j] - \sum_{1\le k\le i-1} a[i,k]*a[k,j]$$

-S'il est sur la diagonale ou en-dessous,on le calcule par la formule:

$$\alpha ij=(a[i,j] - \sum_{1\leq k\leq j-1} a[i,k]*a[k,j])*1/\beta jj$$

mais en un premier temps, on n'évaluera que le premier terme de cette multiplication et on comparera ce terme multiplié par le vv[i] correspondant pour chaque élément de la colonne en question situé sur la diagonale ou en-dessous et on gardera alors le plus grand d'entre eux dans la variable dum etl'indice de la ligne qui le contient sera stocké dans imax.La cellule a[imax,j] est alors le pivot choisi.

- Si le pivot n'est pas situé sur la diagonale(si imax est différent de j) alors on interchange les lignes imax et j afin que le pivot se retrouve sur la diagonale (méthode de pivot partielle) on change la parité de *d et aussi vv[imax] avec vv[j].
- Le vecteur *indx conservera alors cette permutation si elle a eu lieu.
- Si le pivot a[j,j]=0, alors on le met à une valeur limite et la matrice est alors singulière.

- Et finalement on divise par le pivot a[j,j] tous les éléments de la colonne situés en-dessous de la diagonale si on n'est pas à la dernière colonne (puisque celle-ci ne contient que les ßij).

-Et on remonte pour traiter la colonne suivante.

1.4 Algorithme de la fonction Lubksb(a,n,indx,b)

1.4.a Les variables reçues

**a :matrice issue de la décomposition LU effectuée par la fonction Ludcmp.

*indx :vecteur issu de la fonction Ludcmp et contenant les permutations effectuées lors de la décomposition LU de la matrice **a.

n :dimension du système.

b[] :contiendra au départ le second membre du système d'équations à résoudre ,celui-ci nous est fourni par la fonction Userfun(x,alpha,bet) et qui est le terme bet et qui représente la dérivée partielle du logarithme de la fonction de vraisemblance L.à la sortie de Lubksb le vecteur b[] contiendra la solution du système.

1.4.b Les variables internes

ii=0 quand l'indice ii sera mis à une valeur positive,il sera l'index du premier élément non-nul du vecteur b.

1.4.c Algorithme

Le système linéaire à résoudre est:

$$A*x=b;$$

il est équivalent à:

$$(LU)*x=b$$

en tenant compte des permutations issues de la décomposition LU.
Il est aussi équivalent à:

$$L(U*x)=b;$$

qui est équivalent à:

$$U*x=y \text{ et } L*y=b.$$

L*y=b est un système diagonale inférieur,pour le résoudre on va faire la substitution du haut vers le bas suivant la formule:

$$y[1]=b[1]/\alpha 11;$$
$$y[i]=(b[i] - \sum_{1\leq j\leq i-1} \frac{\alpha ij*y[j]}{\alpha ii}$$

avec $\alpha ii=1$ et $i=2,3,..,n.$

Mais avant cela, il faut voir si on a fait une permutation lors de la décomposition da la matrice

A et interchanger alors b[i] et b[indx[i]] puisque le vecteur indx[i] contient le numéro de la ligne interchanger avec la ligne i.

Il reste à résoudre le système U*x=y;U étant une matrice diagonale supérieure,pour résoudre ce système il suffit de faire la substitution de bas en haut suivant les formules:

$$x[n]=y[n]/\beta nn;$$
$$x[i]=(y[i]-\sum_{i+1\leq j\leq n}\beta ij*x[j])*1/\beta jj \quad \text{avec } i=n-1,n-2,\ ,1.$$

1.5 Algorithme de la fonction Mnewt(ntrial,x,n,tolx,tolf)

1.5.a les variables reçues

ntrial:le nombre d'itération de Newton-Raphson pour trouver la solution du système d'équations non linéaires.

x :vecteur qui contiendra au départ les valeurs initiales pour démarrer le processus de résolution de Newton-Raphson et à la fin contiendra la solution du système d'équations non linéaires si le processus converge.

n :dimension du système.

tolx :est la tolérance de x.

tolf :est la tolérance de f[],-f[] étant le vecteur qui contient les équations du système à résoudre.

1.5.b Les variables internes

*indx :vecteur qui sert à mémoriser les permutations.

errf :représente la somme des valeurs absolues des magnitudes des fonctions f[i] qui constituent le système.

errx :représente la somme des valeurs absolues des corrections δxi=bet[i].

*bet :vecteur qui contient au départ les équations du système à résoudre au point de départ x[] et la fin de chaque itération la correction δxi à apporter à la solution.

**alpha:matrice qui contient les dérivées partielles des fonctions qui constituent le système à résoudre.dans notre cas puisque le système à résoudre est constitué de dérivées partielles du logarithme de la fonction de vraisemblance **alpha contiendra des dérivées secondes du logarithme de la fonction de vraisemblance.

- on commence par allouer de la place mémoire à *indx,*bet et **alpha.

- La fonction Mnewt utilise ntrial itérations commençant par une valeur x[1..n].

- Elle appelle la fonction Userfun afin de lui fournir la matrice alpha hessienne ainsi que le vecteur bet.

- si errf est inférieure ou égale à la tolérance tolf fixée alors Mnewt est terminée et x[1..n] constitue la solution.

- On résoudra le système linéaire au point x[1..n] par les fonctions Ludcmp et Lubksb qui nous fournissent la correction à apporter à la solution

c-à-d le bet[1..n] si le système admet une solution.

- On calcule errx,et on fait la mise à jour de la solution obtenue à l'itération précédente.Et si errx est inférieure ou égale à la tolérance tolx fixée alors x[1..n] représente la solution recherchée et Mnewt est terminée;sinon on passe à l'itération suivante.

- on ne doit pas oublier de désallouer les vecteurs alloués au départ à la sortie de "Mnewt".

CHAPITRE IV. MANUEL UTILISATEUR

1.Introduction

Ce logiciel permet de :

- créer un fichier de données en format ASCII,
- faire la mise-à-jour de la base de données,
- de calculer les coefficients de la fonction logistique,
- et enfin de calculer le risque de faire un infarctus du myocarde.

2.Session utilisateur

Lorsque vous lancez le logiciel "JAWAD.EXE",il apparaît une interface dans laquelle se trouve essentiellement des logos dont chacun indique la fonction qu'il offre et en-dessous duquel se trouve un mot significatif dont la première lettre brille. Le fait d'appuyer sur l'une de ces lettres vous permettra d'activer la fonction correspondante.

2.1 Mis-à-jour de la base de données

En appyant sur la lettre m (ou M), la première fiche de la base de données vous apparaît à l'écran ainsi que deux menus l'un au-dessus, l'autre au- dessous de la fiche.

Le premier menu vous indique la façon de feuilleter les fiches de la base de données

"PATIENT.DBF" en appuyant soit sur les touches (PgUp) (↑) pour accéder à la fiche précédente, soit sur (PgDn) ou(↓) pour accéder à la fiche suivante.

le deuxième menu vous indique que vous pouvez soit:

- Encoder une nouvelle fiche en appuyant sur la touche fonction "F1" ainsi vous pouvez remplir tous les champs qui se présentent devant vous, la longueur des champs est indiquée en mode d'affichage inverse vidéo. Les champs Fumeurs ,Diabète, Angina pectoris, et hérédité sont limités aux valeurs (1/0) signifiant respectivement (Oui/Non); tandis que les autres demandent des nombres réels comme indiqué dans les champs. A chaque fois que vous voulez encoder une nouvelle fiche vous êtes invité à appuyer sur la touche fonction "F1", et si le champs nom-prénom est vide la fiche n'est pas enregistrée dans la base de données "PATIENT.DBF".

- Modifier la fiche courante en appuyant sur la touche fonction "F2".Le déplacement dans les champs de la fiche courante vous est offert à l'aide de →,↑,←,↓et Aussi la touche "Enter" vous permet de valider les changements et de vous déplacez d'un champ à l'autre de la fiche courante.

- Supprimer la fiche courante en appuyant sur la touche fonction "F3" , un message d'une demande de confirmation ou d'annulation de la suppression vous est proposé.

- la touche Esc vous permet de sortir de la mis-à-jour et une seconde frappe sur cette touche vous fera quitter le logiciel JAWAD.

Création du fichier ASCII

En appuyant sur la touche f (ou F) on vous demande d'entrer le nom du fichier ASCCI à créer, laissez-le à "PATIENT" puisque c'est le seul fichier de la base de données dont le programme "statis" de calcul des coefficients en tient compte.

REMARQUE IMPORTANTE:

A la fin de toute session de Mise-à-jour, il ne faut pas oublier de créer le fichier ASCII "PATIENT.DAT" en appuyant sur f(ou F) comme indiqué ci-dessus, sinon seule la base de données "PATIENT.DBF" est mise à jour et non pas le fichier "PATIENT.DAT" nécessaire au calcul des coefficients de la fonction logistique.

2.3 Calcul des coefficients de la fonction logistique

En appuyant sur la touche c (ou C),vous déclenchez automatiquement le calcul des coefficients des facteurs à risque de la fonction logistique. Lorsque le calcul est terminé ,appuyez sur une touche pour revenir au menu principal, et éventuellement vous pouvez consulter ces coefficients dans le fichier "RESULTAT.dat" comme indiqué dans un message à l'écran.

2.4 Calcul du risque de faire un infarctus du myocarde

Si vous appuyez sur r (ou R), un message vous indique de rentrer les données du patient pour lequel vous voulez calculer le risque, à la fin de l'encodage des données du patient, le risque de faire un infarctus du myocarde est affiché automatiquement à l'écran et en appuyant sur une touche on revient au menu principal.

3. Recommandations

Ce logiciel est implémenté sur PC IBM, est compatible en langage Dbase et compilé par le compilateur "Clipper".

Le programme Statis qui calcule les coefficients est implémenté en langage "TURBOC".Lorsque vous désirez augmenter le degré de précision, vous devez changer les tolérances tolf et tolx ainsi que ntrial qui est le nombre d'itération (au début ce dernier ne doit pas être élevé mais quand on atteint la certitude que le système converge il faudra changer la valeur de ntrial à une valeur supérieure parfois à une valeur supérieure à 150).

Dans le programme statis il y a aussi possibilité d'imprimer les bet[j] qui contiennent les équations du système à résoudre. Lorsque les bet[j] s'approchent de zéro au degré de précision fixé cela vous indique que le système converge.Toutes ces

manipulations doivent être faites sur TURBOC. Bien après il faut recompiler et linker le programme statis et transférer statis.exe dans le répertoire contenant le logiciel JAWAD.

4.Fichiers nécessaires pour l'exécution du logiciel JAWAD

La liste des fichiers nécessaires pour l'exécution du logiciel JAWAD est la suivante:

DONPAT.DAT

FICHRISQ.DAT

PATIENT.DAT

RESULTAT.DAT

DONPAT.DBF

PATIENT.DBF

TAMP.DBF

PATIENT.NDX

PATIENY.NTX

JAWAD.EXE

RISQUE.EXE

STATIS.EXE

Installation:

créez un répertoire appelé INFARCTUS et y insérez les fichiers cités ci-dessus

```
#include <alloc.h>
#include <stdio.h>
void nrerror(char[]) ;
float *vector(int,int);
double *dvector(int,int);
int *ivector(int,int);
float **matrix(int,int,int,int);
double **dmatrix(int,int,int,int);
void free_vector(float*,int);
void free_dvector(double*,int);
void free_ivector(int*,int);
void free_matrix(float**,int,int,int);
void free_dmatrix(double**,int,int,int);
void exit(int);
void nrerror(error_text)
char error_text[];

/*standard error handler*/
{
void exit();
fprintf(stderr,"run time error....\n");
fprintf(stderr,"%s\n",error_text);
fprintf(stderr,"...now exiting to system...\n");
exit(1);
}

double *dvector(nl,nh)
int nl,nh;
/* allocate a double vector with range [nl..nh]*/
{  double *v;

   v=(double *)malloc((unsigned) (nh-nl+1)*sizeof(double));
   if (!v) nrerror("allocation failure in dvector()");
   return v-nl;
 }


float *vector(nl,nh)
int nl,nh;
/* allocate a float vector with range [nl..nh]*/
{  float *v;

   v=(float *)malloc((unsigned) (nh-nl+1)*sizeof(float));
   if (!v) nrerror("allocation failure in vector()");
   return v-nl;
 }


 int *ivector(nl,nh)
 int nl,nh;

 /*allocates an  int vector with range [nl..nh]*/

{int *v;
 v=(int *)malloc((unsigned) (nh-nl+1)*sizeof(int));
 if (!v) nrerror("allocation failure in ivector()");
 return v-nl;
 }

float **matrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch ;

/*allocates a float matrix with range [nrl..nrh][ncl..nch]*/
{
```

```
 int i;
 float **m;
 /*allocatzs pointer to rows*/
 m=(float **) malloc((unsigned) (nrh-nrl+1)*sizeof(float *));
 if (!m) nrerror("allocation failure 1 in matrix()");
 m -= nrl;

/*allocate rows and set pointer to them*/
 for (i=nrl;i<=nrh;i++)
   { m[i]=(float *) malloc((unsigned) (nch-ncl+1)*sizeof(float));
     if (!m[i]) nrerror("allocation failure 2 in matrix()");
     m[i] -= ncl;
   }
 /*return to array of pointers to rows*/
 return m ;
}


double **dmatrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch ;

/*allocates a double matrix with range [nrl..nrh][ncl..nch]*/
{
 int i;
 double **m;
 /*allocatzs pointer to rows*/
 m=(double **) malloc((unsigned) (nrh-nrl+1)*sizeof(double *));
 if (!m) nrerror("allocation failure 1 in dmatrix()");
 m -= nrl;

/*allocate rows and set pointer to them*/
 for (i=nrl;i<=nrh;i++)
   { m[i]=(double *) malloc((unsigned) (nch-ncl+1)*sizeof(double));
     if (!m[i]) nrerror("allocation failure 2 in dmatrix()");
     m[i] -= ncl;
   }
 /*return to array of pointers to rows*/
 return m ;
}

void free_dvector(v,nl)
double *v;
int nl;
/*frees a double vector allocated by dvecor()*/
{ free((char*) (v+nl));}

void free_vector(v,nl)
float *v;
int nl;
/*frees a float vector allocated by vecor()*/

{ free((char*) (v+nl));}


void free_ivector(v,nl)
int *v,nl;
{free((char*) (v+nl));}

void free_matrix(m,nrl,nrh,ncl)
float **m;
int nrl,nrh,ncl;
{int i;
for(i=nrh;i>=nrl;i--) free((char*) (m[i]+ncl));
free((char*) (m+nrl));
```

```
}


void free_dmatrix(m,nrl,nrh,ncl)
double **m;
int nrl,nrh,ncl;
{int i;
for(i=nrh;i>=nrl;i--) free((char*) (m[i]+ncl));
free((char*) (m+nrl));
}
```

```
#include <stdio.h>
#include <conio.h>
#include <dos.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <stddef.h>
#include <fcntl.h>
#include <io.h>
#include "nrutil.c"



int lect(char*);
#include "lect.c"
void mnewt(int,int,double[],float,float);
#include "mnewt.c"
int ecriture(char*,double[]);
#include "ecriture.c"
int ntrial= 5;
float tolx =1.0e-6;
float tolf =1.0e-6;
int n =   3;
double x[11];
float **mat1 ;
int np,i,j;
char *param_file = "patient.dat",*res="resultat.dat";


void main()
{
 void mnewt();

 x[1]=0.0;
 x[2]=0.0;
 x[3]=0.0;
 x[4]=0.0;
 x[5]=0.0;
 x[6]=0.0;
 x[7]=0.0;
 x[8]=0.0;
 x[9]=0.0;
 x[10]=0.0;
 lect(param_file);
 mnewt(ntrial,n,x,tolx,tolf);
 ecriture(res,x);
 free_matrix(mat1,1,np,1);

 }
```

```
extern float **mat1;
extern int n,np;
extern char *param_file;

int lect(char *par_file)
   { FILE *param;
     float mat2[1500][10] ;
    char ligne[100],c[100],d[100];
    int  i,j,k,l,return_value=1;
      unsigned pos[]={0,3,5,12,20,22,28,35,37},len[]={2,1,6,7,1,5,6,1,1};


       strcpy(par_file,param_file);
       if((param=fopen(par_file,"rt"))==NULL)
         {
          gotoxy(3,3);
          normvideo();
          cprintf("\n appuyer sur une touche pour continuer\n");
          return_value=0;

          }
      else
          {normvideo();
           gotoxy(3,3);
           fgets(ligne,sizeof(ligne),param);
           gotoxy(3,4);

           i=0;
           while(!feof(param))
                {
                  /*retrait de \n au bout de la ligne */
                  strcpy(c,ligne);
                  for(k=0; c[k] !='\n';k++) ;
                  c[k] ='\0';


                 /*  printf("%s\n",c);*/
                  for(j=1;j<=n-1;j++)
                     { strncpy(d,c+pos[j-1],len[j-1]);
                       mat2[i][j] = atof(d);
                       for(l=0;l<100;l++) d[l] = '\0';
                      }
                  fgets(ligne,sizeof(ligne),param);

                  i++;
                 }/*while*/
           np = i;
           for (i=0;i<=np-1;i++) mat2[i][0] =1.0;
           fclose(param);
           }/*else*/
           mat1=matrix(1,np,1,n);

           for(i=1;i<=np;i++)
             for(j=1;j<=n;j++)
             { float inter=0.0;
               inter = mat2[i-1][j-1];
               mat1[i][j]=inter;

              /* printf("mat1[%d][%d] =%6.2f\n",i,j,mat1[i][j]);*/
              }

    return(return_value);
   }/*fin lecture*/
```

```
#define SQR(a) ((a)*(a))
extern  int n,np;
extern  float **mat1;


void userfun(x,alpha,bet)
double x[],**alpha,*bet;
{       int     i,j,k,*z,*ivector();
        double  *xsi,*gam1,*gam2,*dvector();
        void free_dvector();

        gam1   = dvector(1,n);
        gam2   = dvector(1,n);
        xsi    = dvector(1,np);
        z      = ivector(1,np);

        for(i=1;i<=n;i++) {gam1[i]=gam2[i]=0.0;}
        for(i=1;i<=np;i++){xsi[i]=0.0;
                           z[i]  =0;
                          }
/* calcul de xsi == produit de   mat1 et x */
        for(i=1;i<=np;i++)
           {
           for(j=1;j<=n;j++) xsi[i] = xsi[i] + mat1[i][j]*x[j];
           }

    /* calcul de z[i] qui classe l'individu dans l'un des groupes π1 ou π2 */
        { double inter=0.0;

           for(i=1;i<=np;i++)
           {
           inter = exp(xsi[i]);
           if (1/(1+inter) >= inter/(1+inter)) z[i] =1;
             else z[i] =0;
           }
         }

    /* calcul de beta == δlogl/δxj maximum de vraissemlance */

         for(j=1;j<=n;j++)
          {
           for (i=1;i<=np;i++)
            {
              if ( z[i] == 0 ) gam1[j] = gam1[j] + mat1[i][j]/(1+exp(xsi[i]));

                else gam2[j] = gam2[j] + mat1[i][j]*exp(xsi[i])/(1+exp(xsi[i]))
;

            }

           bet[j] = gam2[j] - gam1[j];


           }


    /* calcul de la matrice alpha[1..n][1..n] hessienne */
        for(j=1;j<=n;j++)
        {
           for(k=1;k<=n;k++)
              {
               for(i=1;i<=np;i++)
                  {
alpha[j][k] = alpha[j][k] - mat1[i][j]*mat1[i][k]*exp(xsi[i])/SQR(1+exp(xsi[i]))
```

```
;
                    }

                }
            }
            free_dvector(gam1,1);
            free_dvector(gam2,1);
            free_dvector(xsi,1);
            free_ivector(z,1);

}/*userfun*/
```

```
#define tiny 1.0e-20

void ludcmp(a,n,indx,d)
int n,*indx;
double **a,*d;

   {
     int i,imax,j,k;
     double big,dum,sum,temp;
     double *vv,*dvector();
     void nrerror(),free_dvector();
     vv = dvector(1,n);
     *d=1.0;
     for(i=1;i<=n;i++)
      {
        big=0.0;
        for(j=1;j<=n;j++)
          if ((temp=fabs(a[i][j])) > big )  big = temp;
        if (big == 0.0) nrerror("singular matrix in ludcump");
        vv[i]= 1.0/big;
       }
     for(j=1;j<=n;j++) {
        for(i=1;i<j;i++) {sum =a[i][j];
                          for(k=1;k<i;k++) sum -= a[i][k]*a[k][j];
                          a[i][j] = sum;
                         }
                         big=0.0;

                         for(i=j;i<=n;i++) {sum = a[i][j];
                                            for(k=1;k<j;k++)
                                               sum -= a[i][k]*a[k][j];
                                            a[i][j] = sum ;
                                            if ((dum=vv[i]*fabs(sum)) >= bi
                                             {big=dum;
                                               imax=i;
                                              }
                                           }
                        if (j != imax) {for(k=1;k<=n;k++)
                                          { dum         = a[imax][k];
                                            a[imax][k] = a[j][k] ;
                                            a[j][k]     = dum ;
                                          }
                                        *d = -(*d);
                                        vv[imax] = vv[j];
                                       }
                        indx[j] = imax;
                        if( a[j][j] == 0.0 ) a[j][j] = tiny  ;
                        if( j != n ) {dum =1.0/(a[j][j]) ;
                                      for(i=j+1;i<=n;i++) a[i][j] *= dum;
                                     }
                      }
     free_dvector(vv,1);
   }   /* fin ludcump*/
```

```
void lubksb(a,n,indx,b)
double **a, *b;
int n,*indx;
{
 int i,ii=0,ip,j;
 double sum;

 for (i=1;i<=n;i++)
    { ip = indx[i];
      sum = b[ip];
      b[ip] = b[i];
      if (ii)
          for (j=ii;j<=i-1;j++) sum -= a[i][j] * b[j];
          else if (sum) ii = i;
          b[i] = sum;
    }
 for (i=n;i>=1;i--)
    {sum = b[i] ;
     for (j=i+1;j<=n;j++) sum -= a[i][j]*b[j];
     b[i] =sum/a[i][i];
    }
 }
```

```
#define freereturn {free_dmatrix(alpha,1,n,1);/*for(i=1;i<=n;i++) printf("bet[
]=%8.6lf\n",i,bet[i]);*/free_dvector(bet,1);free_ivector(indx,1);return;}
void userfun(double[],double**,double[]);
#include "userfun.c"
void ludcmp(double**,int,int*,double*);
#include "ludcmp.c"
void lubksb(double**,int,int*,double*);
#include "lubksb.c"
void mnewt(int ntr,int n,double x[],float tolx,float tolf)



  {
    int k,i,*indx;
    double d,errx,errf,*bet,**alpha,*dvector(),**dmatrix();
    void userfun(),ludcmp(),lubksb(),free_dvector(),free_dmatrix(),free_ivecto
);

    indx = ivector(1,n);
    bet  = dvector(1,n);
    alpha= dmatrix(1,n,1,n);
    for(i=1;i<=n;i++)
      for(k=1;k<=n;k++) alpha[i][k]=0.0;

    for (k=1;k<=ntr;k++)
       {
        userfun(x,alpha,bet);
         errf = 0.0;
         for (i=1;i<=n;i++) errf += fabs(bet[i]);
        /* printf("errf=%lf\n",errf);*/
         if (errf <= tolf) freereturn
         ludcmp(alpha,n,indx,&d);
         lubksb(alpha,n,indx,bet);
         errx = 0.0;
         for (i=1;i<=n;i++)
            { errx += fabs(bet[i]);
              x[i] += bet[i];
            }
         if (errx <= tolx)  freereturn
        }
        freereturn

  }/*fin de mnewt*/
```

```
extern int  n;
extern char *res;

int ecriture(char *fic,double x[11])

{FILE *param;
 int return_value = 1,j;
 strcpy(fic,res);
 if( (param = fopen(fic,"wt+")) == NULL )
   {return_value = 0;printf("return_value = %d\n",return_value);}
   else{normvideo();
        for (j=1;j<=n;j++) fprintf(param,"%11.5lf\n",x[j]);
        fclose(param);
        }
 return(return_value);
 }
```

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int patdon(char*,double[]);
int coef(char*,double[]);
int ecrisq(char*,double);
double zz[10],yy[10],r=0.0,xsi=0.0;
char *res="resultat.dat",*param_file="donpat.dat",*dd="fichrisq.dat";



/* cette fonction charge dans un tableau y[10] les données du patient qui
   se trouvent  dans le fichier donpat.dat*/

int patdon(char *par_file,double y[10])
   { FILE *param;
    char ligne[100],c[100],d[100];
    int l,k,j,return_value=1;
      unsigned pos[]={0,3,5,12,20,22,28,35,37},len[]={2,1,6,7,1,5,6,1,1};


       strcpy(par_file,param_file);
       if((param=fopen(par_file,"rt")) != NULL)
           {fgets(ligne,sizeof(ligne),param);

                  /*retrait de \n au bout de la ligne */
                  strcpy(c,ligne);
                  for(k=0; c[k] !='\n';k++) ;
                  c[k] ='\0';

                /*  printf("%s\n",c);*/

                  for(j=1;j<=9;j++)
                     { strncpy(d,c+pos[j-1],len[j-1]);
                       y[j] = atof(d);
                       for(l=0;l<100;l++) d[l]='\0';
                      /* printf("y[%d] =%6.2f\n",j,y[j]);*/
                      }

            y[0]=1.0;
            fclose(param);
         }
    return(return_value);
    }/*fin patdon*/



/*cette fonction charge les coeficients calcules par statis et
  deposés dans resultat.dat dans un vecteur zz[10]*/

int coef(char *par_file,double z[10])
   { FILE *param;
    char ligne[30],c[30];
    int  j,k,l,return_value=1;


       strcpy(par_file,res);
       if((param=fopen(par_file,"rt")) != NULL)
          {
           fgets(ligne,sizeof(ligne),param);
           j=0;
```

```
            while(!feof(param))
                 {
                   /*retrait de \n au bout de la ligne */
                   strcpy(c,ligne);
                   for(k=0;c[k] != '\n';k++);
                   c[k] ='\0';
                   z[j]= atof(c);
                   for(l=0;l<30;l++) c[l] = '\0';

                   fgets(ligne,sizeof(ligne),param);

                   j++;
                  }/*while*/
              fclose(param);
/*    for(j=0;j<10;j++) printf("z[%d] = %lf\n",j,z[j]);*/
  }/*if*/


     return(return_value);
    }/*fin coef */



int ecrisq(char *fic,double x)

{FILE *param;
 int return_value = 1;
 strcpy(fic,dd);
 if( (param = fopen(fic,"wt+")) != NULL )
   {
        fprintf(param,"%lf",x);
        fclose(param);
    }
 return(return_value);
 }




void main()
{int /*s,*/j;
 coef(res,zz);
/*for(s=0;s<10;s++)
    printf("zz[%d]=%lf\n",s,zz[s]);*/



patdon(param_file,yy);
/*for(j=0;j<10;j++)
   printf("yy[%d] =%6.2f\n",j,yy[j]);*/
for(j=0;j<10;j++)
   xsi = xsi + yy[j]*zz[j];
r=1-1/(1+exp(xsi));
/*printf("xsi=%lf\nr=%lf\n",xsi,r);*/
ecrisq(dd,r);


}
```

```
* Procédure: JAWAD.PRG
* Date.....: 10/07/90
* Notes....: Programme de redirection des sorties des données
SET STATUS      OFF
SET TALK        OFF
SET HEADING     OFF
SET SCOREBOARD OFF
SET BELL        OFF
SET DATE FRENCH
SET SAFETY      OFF
SET CONFIRM     ON
SET CONSOLE     ON
SET ESCAPE      ON
*SAVE SCREEN
IF FILE ("MICSLOGO.EXE")
  !MICSLOGO
ENDIF
SET COLOR TO
CLEAR
@  0,  0  TO 15, 79 DOUBLE
@  1,  2  TO  4, 77
@  5,  2  TO 14, 77
SET COLOR TO GR+/B
@  2,8  say [ P R O G R A M M E   D E   T R A I T E M E N T   D E   B A S E ]
SET COLOR TO
@  3,6  say "MicroStat Version 1.0 (c) Copyright  Jawad, Louvain-La-Neuve, 1990"
@  6,4  say "┌──────────────────┐    ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
      "
@  7,4  say "| ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ |    ▒▒ Module          ▒▒┐"
      "
@  8,4  say "| Créer le       ┌┘    ▒▒ de Calculs      ▒▒°▒▒"
      "
@  9,4  say "| Fichier de     └┐    ▒▒ Statistiques     ▒▒°▒▒    ┌─:▒▒▒▒▒▒▒▒▒▒▒▒▒▒:─┐"
      "
@ 10,4  say "| Données en       |    ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒    | :▒▒▒▒▒▒▒▒▒▒▒▒▒▒: |"
      "
@ 11,4  say "| Format ASCII     | ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ |──────────────────┘|┌──────────
──────┐"
      "
@ 12,4  say "| à Traiter ...    | ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ |calcul risque ||  Mise à J
our  |"
@ 13,4  say "└──────────────────┘ ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ └──────────────────┘└──────────
──────┘"
@ 06,61 say "= Quitter"
SET COLOR TO GR+/B
@ 06,57 say "Esc"
@ 14, 8 say " F"
SET COLOR TO
@ 14,10 say "ichier "
SET COLOR TO GR+/B
@ 14,28 say " C"
SET COLOR TO
@ 14,30 say "alculs "
SET COLOR TO GR+/B
@ 14,47 say " R"
SET COLOR TO
@ 14,49 say "isque "
SET COLOR TO /W
SET COLOR TO GR+/B
@ 14,62 say " M"
SET COLOR TO
@ 14,64 say "ise à Jour"
SET COLOR TO /W
@ 24, 0 SAY "MicroStat V1.0|(c) Copyright Jawad, Louvain-La-Neuve, 1990. Tous Dr
oits Réservés"
```

```
SET COLOR TO
! 14, 35 SAY ''
SET COLOR TO
IF .NOT. FILE ("PATIENT.NTX")
F .NOT. FILE ("PATIENT.NDX")
  USE PATIENT
  INDEX ON NOM_PREN TO PATIENT
NDIF
JSE PATIENT INDEX PATIENT
O TOP

O WHILE .T.
  SET CONSOLE ON
  SET COLOR TO /W
  @ 24, 0 SAY "MicroStat V1.0|(c) Copyright Jawad, Louvain-La-Neuve, 1990. Tous
roits Réservés"
  SET COLOR TO
  @ 14, 35 SAY ''
  SET CONSOLE OFF
  store inkey() to ans
  do while ans =0
     store inkey() to ans
  enddo
  SET CONSOLE ON
  DO CASE
    CASE ans = 70 .OR. ans = 102
    *fonction FICHIER
      SAVE SCREEN
      USE PATIENT INDEX PATIENT
      GO TOP
      SET CONSOLE ON
      SET COLOR TO
      @ 22,0 SAY SPACE(80)
      @ 23,0 SAY SPACE(80)
      @ 24,0 SAY SPACE(80)
      @ 22,0 TO 24,79
      FICH="PATIENT.DAT"
      FICHIER="PATIENT"
      @ 23,2 SAY "Entrez le nom du fichier à créer: " GET FICHIER PICTURE "!!!!!
!!!"
      READ
      @ 22,0 SAY SPACE(80)
      @ 23,0 SAY SPACE(80)
      @ 24,0 SAY SPACE(80)
      @ 22,0 TO 24,79
      IF LEN(TRIM(FICHIER))<>0
        FICH=FICHIER+".DAT"
      ENDIF
      @ 23,2 SAY "Constitution du fichier: " + TRIM(FICH) + " en cours ..."
      SET CONSOLE OFF
      COPY TO &FICH FIELDS AGE,SPC,DIAB,SPC,CHOL,SPC,TRIG,SPC,FUM,SPC,HDLC,SPC,P
RES,SPC,HERID,SPC,ANG SDF
      *REPORT FORM DONNEES HEADING "FICHIER DE DONNEES - JAWAD 1990" TO FILE &FI
CH
      SET CONSOLE ON
      @ 22,0 SAY SPACE(80)
      @ 23,0 SAY SPACE(80)
      @ 24,0 SAY SPACE(80)
      *@ 06,5 say space(44)
      @ 22,0 TO 24,79
      @ 23,2 SAY "Fichier "+FICHIER+".DAT est constitué. Appuyez sur une touche
..."
```

```
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   RESTORE SCREEN
CASE ans = 82 .or. ans = 114

 SAVE SCREEN
 @ 5,2 TO 14,77
 SET ESCAPE ON
 CLOSE DATA
 USE DONPAT.DBF
 ZAP
 SET COLOR TO
 @ 24,0 SAY SPACE(80)
 @ 6,57 SAY SPACE (13)
 SET COLOR TO I/
 @ 16,25 SAY "VEUILLEZ ENTREZ LES DONNEES DU PATIENT"
 @ 24,15 SAY "F.U.N.D.P | Institut d'Informatique | Tous Droit Reserve"
 SET COLOR TO
 MNOM_PREN = SPACE(34)
 MADRESSE  = SPACE(34)
 MAGE      = 0
 MFUM       = 0
 MDIAB      = 0
 MCHOL      = 0
 MTRIG      = 0
 MANG       = 0
 MHDLC      = 0
 MPRES      =0
 MHERID     = 0

 DO MATSCR
 READ
 IF LEN(TRIM(MNOM_PREN)) <> 0
    APPEND BLANK
    REPLACE NOM_PREN WITH MNOM_PREN
    REPLACE ADRESSE WITH MADRESSE
    REPLACE AGE WITH MAGE
    REPLACE FUM WITH MFUM
    REPLACE DIAB WITH MDIAB
    REPLACE CHOL WITH MCHOL
    REPLACE TRIG WITH MTRIG
    REPLACE ANG WITH MANG
    REPLACE HDLC WITH MHDLC
    REPLACE PRES WITH MPRES
    REPLACE HERID WITH MHERID

    *REMPLISSAGE DU FICHIER DONPAT.DAT
    FII = "DONPAT.DAT"
    SET CONSOLE OFF
    COPY TO &FII FIELDS AGE,SPC,DIAB,SPC,CHOL,SPC,TRIG,SPC,FUM,SPC,HDLC,SPC,
PRES,SPC,HERID,SPC,ANG SDF


    I=16
    DO WHILE I<25
     @ I,0 SAY SPACE(80)
     I= I+1
    ENDDO
    *CALCUL DU RISQUE ET DEPOSITION DU RESULTAT DANS FICHRISQ.DAT
    ! RISQUE.EXE
    USE TAMP.DBF
    ZAP
    APPEND FROM FICHRISQ.DAT SDF
```

```
      CH = TRIM(SUBSTR(RES,1,15))
      *res est un champ de tamp.dbf
      NB = VAL(CH)
      @ 21,24 TO 23,57 DOUBLE
      SET COLOR TO GR+/B
      @ 22,25 SAY "LE RISQUE VAUT: "
      @ 22,40 SAY NB
      SET COLOR TO
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
      CLOSE DATA
      RESTORE SCREEN
      ENDIF


  CASE ans = 77 .OR. ans = 109
    SAVE SCREEN
    DO MISAJOUR
    RESTORE SCREEN
  CASE ans = 67 .OR. ans = 99
   * USE PATIENT INDEX PATIENT
    SET ESCAPE ON
   * CLOSE DATA
    SAVE SCREEN
    @ 5,2 TO 14,77
    @ 06,57 SAY SPACE(13)
    SET COLOR TO I/
   * @ 17,18 TO 21,62
    @ 18,19 TO 20,65 DOUBLE
    @ 19,20 SAY "  CALCUL EN COURS; VEUILLEEZ PATIENTEZ SVP.. "

    ! STATIS.EXE
    @ 19,20 SAY SPACE(42)
    @ 19,20 SAY " Apuiyez sur une touche;VOIR RESULTAT.DAT "
    SET COLOR TO
    SET CONSOLE OFF
    WAIT
    SET CONSOLE ON
   * USE PATIENT INDEX PATIENT
    RESTORE SCREEN
  CASE ans = 27
    RESTORE SCREEN
    RELEASE ALL
    CLOSE DATA
    SET COLOR TO
    CLEAR
    QUIT
 ENDCASE
ENDDO
* Fin du Programme: JAWAD.PRG
```

```
*MATLSCR.PRG MODULE D'ENTREE SORTIE
SET COLOR TO
SET CONFIRM OFF
SET COLOR TO W+
@ 17,0  SAY '┌──────────────────────────────────────────────────────────────────────────────┐
──────────┘'
@ 18,0  SAY '│ Nom Prénom :                                                  Age    :
 [40-65]  '
@ 19,0  SAY '│ Adresse    :                                                  Fumeur:
 Diabète:  '
@ 20,0  SAY '├──────────────────────────────┬──────────────────────────────┬──────────
──────────┘'
@ 21,0  SAY "│ Cholesterol:         ml/dl│Triglycer. :                  │Angina Pector
is:       1/0"
@ 22,0  SAY "│ Hdlc       :         ml/dl│Pression   :           mmhg   │Hérédité
  :       1/0"
@ 23,0  SAY '└──────────────────────────────┴──────────────────────────────┴──────────
──────────┘'
SET COLOR TO
@ 18,15 SAY NOM_PREN
@ 19,15 SAY ADRESSE
@ 18,63 SAY AGE
@ 19,63 SAY FUM
@ 19,77 SAY DIAB
@ 21,15 SAY CHOL
@ 21,41 SAY TRIG
@ 21,71 SAY ANG
@ 22,15 SAY HDLC
@ 22,41 SAY PRES
@ 22,71 SAY HERID
RETURN
```

```
MATLSCR.PRG MODULE D'ENTREE SORTIE
ET COLOR TO
ET CONFIRM OFF
ET COLOR TO W+

 17,0  SAY '┌──────────────────────────────────────────────────────────────────────────────┐
──────────┐'
 18,0  SAY '│ Nom Prénom :                                                  Age    :
[40-65]   │'
 19,0  SAY '│ Adresse    :                                                  Fumeur:
Diabète:  │'
 20,0  SAY '├──────────────────────────────┬──────────────────────────┬──────────────────
──────────┤'
 21,0  SAY "│ Cholesterol:        ml/dl│Triglycer. :              │Angina Pector
s:      1/0│"
 22,0  SAY "│ Hdlc       :        ml/dl│Pression   :          mmhg│Hérédité
 :      1/0│"
 23,0  SAY '└──────────────────────────────┴──────────────────────────┴──────────────────
──────────┘'
ET COLOR TO
 18,15 GET MNOM_PREN
 19,15 GET MADRESSE
 18,63 GET MAGE PICTURE "99"
 19,63 GET MFUM PICTURE "9"          RANGE 0   , 1
 19,77 GET MDIAB PICTURE "9"         RANGE 0   , 1
 21,15 GET MCHOL PICTURE "999.99"
 21,41 GET MTRIG PICTURE "9999.99"
 21,71 GET MANG  PICTURE "9"         RANGE 0   , 1
 22,15 GET MHDLC PICTURE "99.99"
 22,41 GET MPRES PICTURE "999.99"
 22,71 GET MHERID PICTURE "9"        RANGE 0   , 1
RETURN
```

```
* MISAJOUR.PRG  Procédure de mise à jour
* ---------------------------------------
STORE SPACE(34) TO MNOM_PREN
STORE SPACE(34) TO MADRESSE
STORE 0         TO MAGE
STORE 0         TO MFUM
STORE 0         TO MDIAB
STORE 0         TO MCHOL
STORE 0         TO MTRIG
STORE 0         TO MANG
STORE 0         TO MHDLC
STORE 0         TO MPRES
STORE 0         TO MHERID
SET DELETED ON
USE PATIENT INDEX PATIENT
GO TOP
SET COLOR TO
@  5,  2  TO 14, 77
DO WHILE .T.
  SET COLOR TO
  @ 6 ,57 SAY SPACE(13)
  SET COLOR TO I/
  @ 16, 0 SAY SPACE(80)
  @ 16, 0 SAY "Rappel: «"+CHR(24)+" PgUp»: Fiche Précédente | «"+CHR(25)+" PgDn»
: Fiche Suivante"
  SET COLOR TO
  DO MATLOOK
  SET COLOR TO I/
  @ 24, 0 SAY SPACE(80)
  @ 24, 0 SAY "«F1»: Encodage  | «F2»: Modification  | «F3»: Suppression | «Esc»
: Quitter M.A.J"
  SET COLOR TO
  STORE NOM_PREN TO MNOM_PREN
  STORE ADRESSE  TO MADRESSE
  STORE AGE      TO MAGE
  STORE FUM      TO MFUM
  STORE DIAB     TO MDIAB
  STORE CHOL     TO MCHOL
  STORE TRIG     TO MTRIG
  STORE ANG      TO MANG
  STORE HDLC     TO MHDLC
  STORE PRES     TO MPRES
  STORE HERID    TO MHERID
  SET CONSOLE OFF
  store inkey() to ans
  do while ans =0
     store inkey() to ans
  enddo
  SET CONSOLE ON
  DO CASE
    CASE ans = 28
    *fonction encodage

      STORE SPACE(34) TO MNOM_PREN
      STORE SPACE(34) TO MADRESSE
      STORE 0         TO MAGE
      STORE 0         TO MFUM
      STORE 0         TO MDIAB
      STORE 0         TO MCHOL
      STORE 0         TO MTRIG
      STORE 0         TO MANG
      STORE 0         TO MHDLC
      STORE 0         TO MPRES
      STORE 0         TO MHERID
```

```
   DO MATSCR
   READ
   IF LEN(TRIM(MNOM_PREN))<> 0
     APPEND BLANK
     REPLACE NOM_PREN WITH MNOM_PREN
     REPLACE ADRESSE  WITH MADRESSE
     REPLACE AGE      WITH MAGE
     REPLACE FUM      WITH MFUM
     REPLACE DIAB     WITH MDIAB
     REPLACE CHOL     WITH MCHOL
     REPLACE TRIG     WITH MTRIG
     REPLACE ANG      WITH MANG
     REPLACE HDLC     WITH MHDLC
     REPLACE PRES     WITH MPRES
     REPLACE HERID    WITH MHERID
   ENDIF
 CASE ans = -1
 *fonction modification
   DO MATSCR
   READ
   IF LEN(TRIM(MNOM_PREN))<> 0
     REPLACE NOM_PREN WITH MNOM_PREN
     REPLACE ADRESSE  WITH MADRESSE
     REPLACE AGE      WITH MAGE
     REPLACE FUM      WITH MFUM
     REPLACE DIAB     WITH MDIAB
     REPLACE CHOL     WITH MCHOL
     REPLACE TRIG     WITH MTRIG
     REPLACE ANG      WITH MANG
     REPLACE HDLC     WITH MHDLC
     REPLACE PRES     WITH MPRES
     REPLACE HERID    WITH MHERID
   ENDIF
 CASE ans = -2
 *fonction suppression
   NUM=RECNO()
   SET COLOR TO I/
   @ 18,12 SAY "                                                "
   @ 17,12 TO 19, 57 DOUBLE
   Ok="N"
   SET CONFIRM OFF
   @ 18,14 SAY "Veuillez Confirmer cette action «O/N» ? " GET OK PICTURE "!"
   READ
   SET COLOR TO
   IF (Ok="O")
     DELETE
     PACK
   ENDIF
   IF .NOT. BOF()
     GOTO NUM-1
   ELSE
     GOTO NUM
   ENDIF
 CASE ans = 24 .or. ans = 3
 *fonction positionement -
   SET ESCAPE ON
   IF .NOT. EOF()
     SKIP
    ENDIF
    SAVENUMA=RECNO()
    IF .NOT. BOF() .AND. .NOT. EOF()
      NUMA=RECNO()
    ENDIF
    SET ESCAPE OFF
  CASE ans = 5 .or. ans = 18
```

```
     *fonction positionement +
       SET ESCAPE ON
       IF .NOT. BOF()
         SKIP-1
       ENDIF
       IF .NOT. EOF() .AND. .NOT. BOF()
         NUMA=RECNO()
       ENDIF
       SAVENUMA=RECNO()
       SET ESCAPE OFF
    CASE ans = 27
    *fonction escape
      i=16
      SET COLOR TO
      DO WHILE I<25
        @ i,0 SAY SPACE(80)
        i=i+1
      ENDDO
      EXIT
    ENDCASE
ENDDO
RETURN
```

Bibliographie

Cox D.R, [1970], "Analysis of binary data", Chapman and Hall

Lindgren B.W, [1976], "Statistical theory",
Mac Millan publiching

Printice R.L, [1976], "Use of logistic model in retrospective studies", Biometrics 32, 599-606

Printice R.L and Breslow, [1978], "retrospective studies and failure time models", Biometrika 65, 153-158

Anderson J.A, [1972], "Separate sample logistic discrimination", Biometrica 59, 19-35

Anderson J.A ,[1973], "Logistic discrimination whith medical application", 1-13, New-york: Academic Press

Albert A, [1980], "Discrimination logistique", Analyse discriminante, Edité par L.Bragard

C.Rumeau, G.beart et R.Padieu [1981] "Méthode en épidémiologie" édité par Flamarion Médecine Sciences

P.Lassaux et R Theodore, "Analyse Numérique Matricielle appliqué à l'art de l'ingénieur"

G.assmann and Schulte [1986] "Procam-Trial" Panscientia Verlag Hedingen/Zürich.